OPTIMIZATION OF MACHINING STRATEGY AND PROCESS PLANNING OF COMPLEX GEOMETRY

By

AMRO M. FIKRY YOUSSEF

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfillment of the requirements
for the Degree of

Doctor Of Philosophy

McMaster University

OPTIMIZATION OF MACHINING STRATEGY AND PROCESS PLANNING OF COMPLEX GEOMETRY

Doctor Of Philosophy McMaster University
Hamilton, Ontario

TITLE: Optimization of Machining Strategy and Process Planning of Complex Geometry

AUTHOR: Amro M. Fikry Youssef, M.Sc.

SUPERVISOR: Dr. M. A. Elbestawi

NUMBER OF PAGES: 190 pages (i-xvii, 1-172)

ABSTRACT

Advances in the aerospace and automotive industry have led to the introduction of many new shapes with complex geometries. These shapes are typical of turbine blades, as well as molds and dies. The manufacturing of these geometries consists of automatic tool path generation for multi-axis computer numerically-controlled machining centers. The conformity of the manufactured geometry to the design requirements depends on several factors. These involve cutting tool selection, positioning of the cutting tool, as well as the avoidance of collision and gouging between the cutting tool and the workpiece.

This dissertation presents a solid modeling-based approach for tool geometry selection, the simulation and optimization of five-axis tool path generation, as well as gouge and collision avoidance for manufacturing sculptured surfaces. A methodology for tool path generation and checking for the flank-milling of operation is also developed.

An optimization algorithm is developed to re-distribute tool path points according to their effect on the geometry of the manufactured surface. A new algorithm to determine tool inclination is also developed based on an effective tool profile and curvature matching techniques. A cusp height control approach is developed based on feasible tool inclinations, and step-over distance evaluation. An algorithm for tool path generation and gouge as well as collision avoidance is developed.

A flank milling tool path planning and checking algorithm is developed based on the optimization of the tool path advancement scheme according to the surface topology.

A new multi-stage tool model refinement approach was developed, and integrated with the algorithms that constitute this work.

Experimental validation of the presented algorithms is carried out to test the

cusp height, and smoothness of the tool orientation transition. A comparison between the presented algorithms and state of the art CAM commercial software is also presented. The validation results showed that the developed approach can give more accurate results.

The results of this work are intended to fill the shortcomings encountered in the previous research on five-axis machining of sculptured surfaces. These are: cutting tool selection, tool path points optimization, cusp height control, tool position and orientation optimization, and flank-milling.

To My Father, Mother, and My wife Doaa.

ACKNOWLEDGMENTS

The author would like to express his sincere gratitude to his research supervisor Professor Mohamed Elbestwai for his guidance, and support throughout the course of this research. Sincere appreciation is also expressed to all the members of the supervisory committee Dr. John. Vlachopoulos, Dr. Samir Ziada, and Dr. Stephen Veldhuis.

Special thanks to Dr. Tahani Elwardani for her continuing help and support all the time. Special thanks also goes to my friend Diaa ElKott for his helpful discussions and in encouragements.

Special Thanks to all the staff and colleagues at the Mechanical Engineering Department and McMaster Manufacturing Research Institute for their help, specially to Dr. Eugene Ng for his help in the preparation of this work.

Finally, the author is very grateful for the support of the Egyptian Ministry Of defense for making this research possible.

NOMENCLATURE

λ_{incl}	Tool inclination angle[rad.].
$\mathbf{t}$	Unit tangent vector.
$\overline{S}_{rot}$	Tool rotation vector.
$\overrightarrow{S}_{Coll}$	Collision deviation evaluation vector.
$\overrightarrow{S}_{dev}$	Gouge deviation evaluation vector.
ϕ_{base}	Tool base inclination angle.
ϕ_{flank}	Tool flank inclination angle.
ψ_e	Fillet end angle[rad.].
ψ_i	Instantaneous contact angle[rad.].
ψ_s	Fillet start angle[rad.].
ψ_{Coll}	Collision compensation angle[rad.].
ψ_{gouge}	Gouge compensation angle[rad.].
ψ_{tool}	tool flank compensation angle [rad.].
Ψ_{tool}	The tool profile function.
$\triangle k_n$	Curvature change.
H	Total Tool Length.
H_{flank}	Length of the tool flank.
k_{max}	Maximum principal curvature.

k_{min}	Minimum principal curvature.
k_n^u	Normal curvature in u direction.
k_n^v	Normal curvature in v direction.
K_{gauss}	Gaussian curvature.
K_{mean}	Mean curvature.
L_{gouge}	Radial gouge distance.
M_0, M_1, M_2	Tool contact points in flank milling.
M_{cc}	Tool/workpiece contact point.
M_{Coll_wp}	Coll evaluation point on workpiece intersection curve.
M_{dev_wp}	gouge evaluation point on workpiece intersection curve.
M_{fillet}	Fillet-section center point.
M_{rot}	Tool rotation point.
N	Normal vector to the surface S($u.v$).
R_1	The tool radius.
R_2	fillet/insert radius.
R_m	Major tool radius of conical milling cutter .
R_{1_m}	Modified tool radius(gouge).
s	arc length of parametric curve.
S_u	partial derivatives of the surface S in the directions u.

S_v	partial derivatives of the surface S in the directions v.
T_{rot}	Tool rotation plane.
u, v	Independent variables in parametric space.
x, y, z	Independent variables in Cartesian space.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 Motivation and Background

Advances in the aerospace industry have created several engineering challenges. These challenges include the introduction of complex geometry and difficult to machine alloys. Monolithic parts are utilized in the aerospace industry to reduce stresses, prolong service life, and reduce weight. The fabrication of parts with these features is associated with high cost, induced by higher machining time and material cost, as compared to aluminum or hardened steel.

In a competitive industrial environment the production cost must be reduced. This can be achieved through the reduction of the number of manufacturing processes, machining time and by producing the first part right.

The complex part geometries require the use of sophisticated computer aided manufacturing (CAM) tools. The CAM tools must have the capability to handle multi-axis programming, detection and avoidance of tool gouging and collision, in addition to proper tool selection. The CAM software should also include the capability of predicting cutting forces, temperature, tool deflection and process dynamics. The use of five-axis milling is preferred to reduce the geometric errors generated during 3-axis machining. However, five-axis milling is a very complex process that involves several tasks (e.g.; tool orientation, gouge and collision avoidance). Ball nose end mills are widely used in multi-axis milling to facilitate the tool orientation process. This may result in geometric errors in the form of scallops. These errors need further finishing operations to reach the desired surface finish. However, to reduce the geometric errors, a very dense tool path should be created. This may result in very lengthy CNC programs that are difficult to handle by the machine tool controllers. To improve the manufacturing process efficiency, other types of milling cutters (flat, filleted, conical, etc) are introduced to five-axis milling. The new cutter geometries led to the reduction of CNC program size, while reducing the surface error by better matching the tool geometry with the part surface. On the other hand, the cutter positioning and gouging problem becomes more complex. Several checks became essential to detect and avoid undercuts and back gouging.

Two major limitations were observed in the work reported in the literature, as well as in commercial CAM packages. First, the tool selection is usually performed before the tool path is generated. This leads to tool paths that may fail to produce the desired surface geometry. The second limitation is that the tool path is generated without considering gouging and collisions, followed by the gouge and collision detection procedures. This leads to very lengthy calculations to optimize the whole tool path in a iterative manner. This research aims at developing a strategy that integrates tool selection, cutter path planning with gouge detection and collision avoidance to optimize the multi-axis milling processes for the production of complex-geometry parts.

1.2 Scope of the Work

Based on the limitations in the work reported in the literature, and the motivation discussed in the previous section, the objectives of this research are;

- A strategy for five-axis tool path planning and optimization based on non-isoparametric tool path planning methods.
- Tool path evaluation strategies based on part surface topology. This requires the ability to use different tool geometries, which requires the development of a generic tool model.
- Tool positioning strategy based on a solid modeling environment.
- Tool gouge detection and avoidance as well as collision detection and avoidance algorithms.
- Evaluate various cutting strategies, such as tool flank milling of ruled surfaces and point milling of complex free form surfaces.
- A multi-stage tool selection algorithm based on the integration of tool selection with the gouge and collision avoidance modules.

The downstream application of this work may include the investigation of production of integrated blade rotors (IBRs) as an example of a complicated design.

The class of free−form surfaces under investigation is limited to the following;

1. Bounded surfaces. The workpiece surface under investigation should be self contained within its geometrical boundaries.

2. Quadri−lateral surfaces enclosed by four sides.

3. No trimming/stitching may be permitted within the surface boundaries.

Figure 1.1, represents an overview for the developed system.

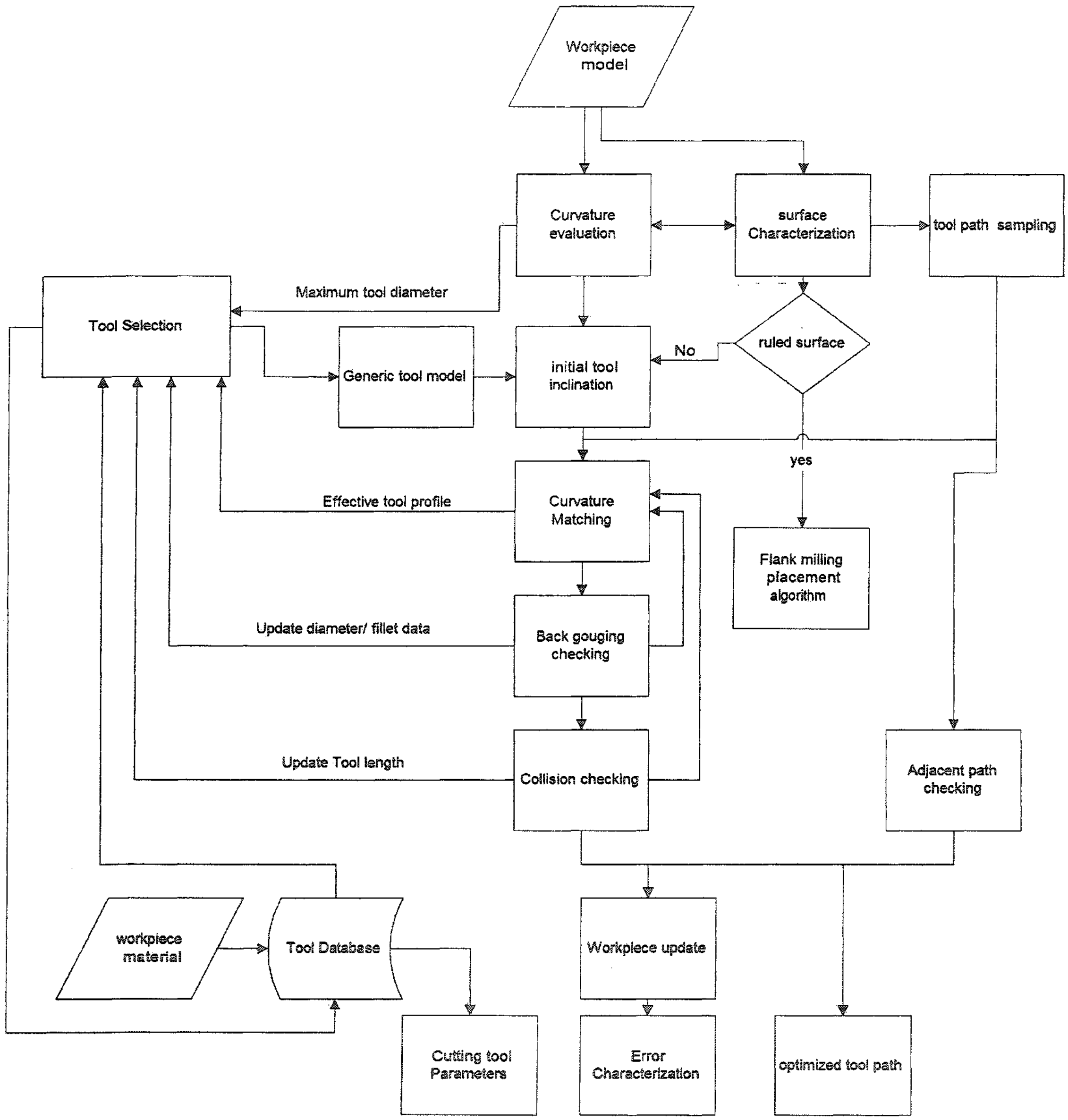


Figure 1.1: Scope of research overview

1.3 Thesis Outline

This thesis is compromised of ten chapters and one appendix. Throughout the presentation of this thesis, several illustrations adapted from the developed system are presented. These illustrations present the results of each phase.

Chapter two is an overview of the relevant research work. These sections covers modeling techniques, tool path planning strategies, milling simulation techniques, tool positioning in multi-axis machining, collision and gouge detection and avoidance and finally flank milling of ruled surfaces.

Chapter three presents the approach and development for the workpiece topology evaluations. This includes the review of different criteria used to investigate this problem, the evaluation of the curvature and curvature changes to pinpoint the important areas within workpiece surface.

In chapter four, the development of a generic tool model is presented.

Chapter five presents the development of a feasible tool inclination range evaluation process. In addition, to the presentation of a step-over evaluation process.

Chapter six presents the developments of a tool path generation and checking module. The chapter also includes the presentation of the developments on gouge detection and avoidance as well as collision detection and avoidance methodology.

In chapter seven, the development of a flank milling tool path planning and checking strategy is presented. The flank milling of the developable surface is also investigated. Tool path errors are checked, and the optimized tool path algorithms developments are presented.

Chapter eight, summarizes the tool selection process. The relationship between the tool model refinement procedures and the other modules of this work is also presented.

Chapter nine, presents the verification of the methodology of the developed work. This includes the comparison between the developed system results and the commercial CAM package output. In addition to this, the experimental results obtained from machining tests using the output of the current developed system are also presented.

Chapter ten, presents a summary of the results of this work as well as suggestions for future research in this area.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter describes the main aspects related to 5−axis milling simulation and optimization.

2.2 Computer Representation of Physical Models

In the computer integrated manufacturing environment (CIM), the modeling of the workpiece and/or the tool is an essential part of the process. The degree of the complexity of the model depends on the desired output and the required accuracy. The modeling techniques can be categorized into **analytical** and **geometrical** methods with subcategories under each of them. In the following sections some of these techniques will be outlined in detail.

2.2.1 Analytical Modeling

The analytical modeling of the physical system is not a new aspect. Before the computer graphics age, all the geometrical primitives had mathematical formulae to represent and solve various problems regarding the relationship between these primitives. Since the main focus of this work is to deal with free–form surfaces the analytical representation of these types of entities will be discussed.

There are two analytical methods that are commonly used for free–form surface representation the implicit and explicit forms. The implicit equations impose certain conditions on the varying locus of a point in the Cartesian space, which takes the form $f(x, y) = 0$, for a point on a planar curve, and $f(x, y, z) = 0$ for a point on a surface. For example in equation (2.1), a sphere is implicitly represented.

$$f(x, y, z) = x^2 + y^2 + z^2 - 1 = 0 \tag{2.1}$$

The explicit equations express the varying value of a point coordinate in terms of the other coordinates. They take the form $y = f(x)$ for planar curves, and $z = f(x, y)$ for surfaces. In this form the equation of the sphere will be as shown in equation (2.2).

$$z = \sqrt{x^2 + y^2 + -1} \tag{2.2}$$

In industrial application, the geometric entities utilized in the product design are not limited to cylindrical, spherical and planar surfaces. Complex shapes are heavily utilized in aerospace and automotive components. These shapes, known as free–form surfaces, are defined as a collection of interconnected and bounded parametric patches that are governed by blending and interpolation formulae (Zeid (1991)).

For free–form surfaces the analytical form required to accurately describe the objects depends on the complexity of the surface and may become very complicated or even impossible to form. Thus parametric representations are widely used in the free–form surface modeling area.

In the parametric representation of the entities, the point coordinates are expressed as a function of independent parameter(s) in the parametric space. These parameters are usually termed (u,v). The value of these variables can be mapped to Cartesian space, for instance the locus of points on a planar curve can take the form as detailed in equation (2.3).

$$\mathbf{P} = \begin{bmatrix} x & y \end{bmatrix}^{\mathbf{T}} = \begin{bmatrix} x(u) & y(u) \end{bmatrix}^{\mathbf{T}} \wedge \mathbf{u_{min}} \leq \mathbf{u} \leq \mathbf{u_{max}} \tag{2.3}$$

while for a three dimensional curve or surface the point will be described by equation (2.4).

$$\mathbf{P} = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathbf{T}} = \begin{bmatrix} x(u,v) & y(u,v) & z(u,v) \end{bmatrix}^{\mathbf{T}} \wedge \mathbf{u_{min}} \leq \mathbf{u} \leq \mathbf{u_{max}} \tag{2.4}$$

Thus, equation (2.5) shows the parametric representation of the sphere.

$$\mathbf{S} = \begin{bmatrix} x(u,v) & y(u,v) & z(u,v) \end{bmatrix}^{\mathbf{T}} = \begin{bmatrix} sin(u)cos(v) & sin(u)sin(v) & cos(u) \end{bmatrix}^{\mathbf{T}} \tag{2.5}$$

where:

$$0 \leq u \leq \pi \wedge 0 \leq v \leq 2\pi$$

In comparison to the analytical representation the parametric representations are *more efficient* in representing space curves and surfaces. For example, an implicit representation of a space curve ($C(x, y, z) = 0$) has to be derived from any of the following cases described in equation (2.6).

$$\{\ C(x,y,z)=0\ \} \Leftarrow \begin{cases} f(x,y,z)=0 \cap g(x,y,z)=0 \\ S(u,v) \cap f(x,y,z)=0 \\ S_1(u_1,v_1) \cap S_2(u_2,v_2) \end{cases} \tag{2.6}$$

whereas, the parametric representation of the space curve is expressed in (2.7):

$$\begin{bmatrix} x(u) & y(u) & z(u) \end{bmatrix}^T \wedge u_{min} \leq u \leq u_{max} \tag{2.7}$$

The three methods of curve and surface representation are compared in table (2.1)(Patrikalakis and Maekawa (2000)).

COMPARISON	EXPLICIT EQ.	IMPLICIT EQ.	PARAMETRIC REP.
Traceability	Easy.	Difficult.	Easy.
Transformation	Difficult.	Difficult.	Easy.
Closed and Multi-valued curves	Difficult.	Easy.	Easy.
Free–form surfaces	Difficult.	Difficult.	Easy.
Composite surfaces	Difficult.	Difficult.	Easy.
Point classification	Easy.	Difficult.	Difficult.
Intersections	Easy.	Difficult.	Difficult.

Table 2.1: Comparison of curve and surface modeling methods

Most CAD systems utilize parametric representation to model complex geometries. Several methods for parametric representation of curves and surfaces have been reported in the literature (e.g. Böhm et al. (1984)). The most widely-utilized are

Bézier, B-Splines, as well as NURBS curves, and surfaces. These representations are discussed in section 2.2.3.

2.2.2 Geometrical Modeling

A model can be defined as a representation that describes the physical, topological and, the response of an object. Several kinds of models exist that serve these purposes. These include physical prototypes, mathematical models, and drawings. Computer models aim to create symbolic representations of the physical world that can be manipulated within the computer environment. The geometric model is the core of computer models. Geometric modeling systems can be categorized as wireframe, surface, and solid modeling systems (Lee (1999)). Table 2.2 compares the major geometric modeling systems.

COMPARISON	WIREFRAME	SURFACE	SOLID
Representation	1. Vertices. 2. Lines.	1. Vertices. 2. Lines. 3. Surface equations. 4. Surface topology data.	1. Vertices. 2. Lines. 3. Surface equations. 4. Surface topology data. 5. Volume properties.
Advantages	1. Compact.	1. Surface Information.. 2. Reasonable storage.	1. Surface and volume Information. 2. More realistic representation. 3.Accurate surface connectivity.
Disadvantages	1. Ambiguous. 2. No topological, or physical properties included.	1. No physical properties included. 2. Surface relations is not always saved.	1. Larger storage.
Applications	1. Drafting and perspective drawings.	1. virtual realism. 2. Tool path planning. 3. Robot motion planning. 4. Geometric inspection.	1. Virtual realism. 2. Tool path planning. 3. Volume-related analysis. 4. Finite Element Analysis.

Table 2.2: Comparison of geometric modeling systems

2.2.3 Free–Form Surfaces Representation

In the following sections the parametric representation of free–form curves and surfaces will be outlined.

2.2.3.1 Bézier Representation of Curves and Surfaces

A Bézier curve of the n^{th} degree is defined using a polygon that approximates its shape. The number of vertices of that polygon is equal to $n+1$. The curve passes through the first, and last vertices of the polygon, and is tangent to its first, and last edges. The shape of the polygon controls the shape of the curve. Therefore, this polygon is known as the *control polygon*, and its vertices as the *control points*. A

Bézier curve point is defined using the Bernstein polynomials that blend the effect of each of the control points on the shape of the curve. Figure 2.1 illustrates a 3^{rd}-degree Bézier curve, and the corresponding Bernstein polynomials.

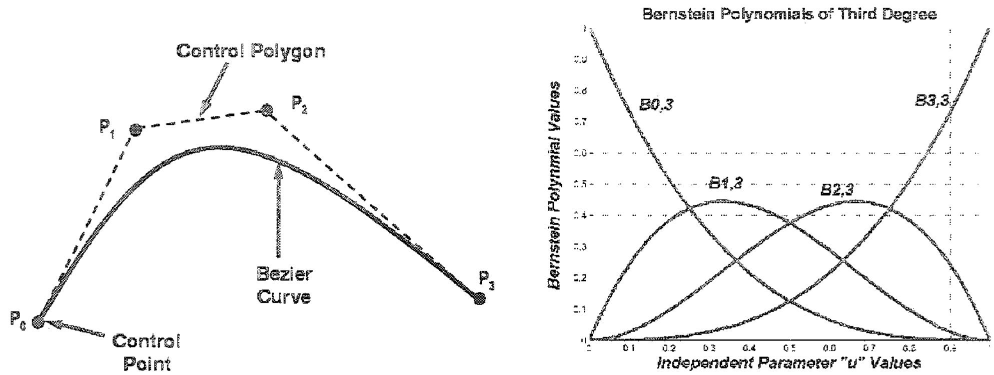


Figure 2.1: Sample Bézier curve and the corresponding Bernstein polynomials

A Bézier curve is defined as in equations (2.8), and (2.9).

$$C(u) = \begin{bmatrix} x(u) & y(u) & z(u) \end{bmatrix}^T = \sum_{i=0}^{n} B_{i,n}(u)\mathbf{P}_i, 0.0 \leq u \leq 1.0 \tag{2.8}$$

$$B_{i,n}(u) = \frac{n!}{i!\,(n-i)!}(1-u)^{n-i}u^i, i = 0, \cdots, n. \tag{2.9}$$

Bézier curves have the following properties of interest (Patrikalakis and Maekawa (2000)):

1. Geometry invariant: Invariance of the shape of the curve under transformations of its control polygon.

2. End points geometric property: Curve *has to* pass through the two end control points, and *has to* be tangent to the first, and last chords of the control polygon.

3. Convex hull property.

Despite the flexibility that Bézier curves offer for shape design, they have the following limitations:

1. The number of control points is related to the curve degree.
2. The global effect of the control points on the curve shape.
3. End points geometric property.

Bézier surfaces extend the Bézier curve definition an extra dimension in the parametric space. A Bézier surface patch of the n^{th} degree in the u-direction, and the m^{th} degree in the v-direction is defined in equation (2.10).

$$S(u,v) = \begin{bmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{bmatrix} = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) B_{j,m}(v) \mathbf{P}_{i,j}, 0.0 \leq u, v \leq 1.0 \tag{2.10}$$

where, $\mathbf{P}_{i,j}$ form the control polyhedron of the Bézier surface patch, $i = 0, \cdots, n \wedge j = 0, \cdots, m$. All properties of Bézier curves have their equivalent counterparts for Bézier surface patches.

2.2.3.2 B-Spline Representation

The formulation of B-Splines was motivated by the two major limitations of the Bézier representation, i.e., global shape control, and the limit on the number of control points. Like a Bézier curve, a B-Spline curve is approximated by the control polygon whose vertices control the shape of the curve. The effects of the control points on the shape of the B-Spline curve are *blended* using the B-Spline basis functions (Patrikalakis and Maekawa (2000)). A B-Spline curve of the p^{th} degree, whose control polygon is defined with $n+1$ control points is formulated as shown in equation (2.11).

$$C(u) = \begin{bmatrix} x(u) & y(u) & z(u) \end{bmatrix}^T = \sum_{i=0}^{n} N_{i,p}(u)\mathbf{P}_i, u_{min.} \le u \le u_{max.} \tag{2.11}$$

where:

$$N_{i,p}(u) = \frac{u - u_i^k}{u_{i+p}^k - u_i^k} N_{i,p-1}(u) + \frac{u - u_{i+p+1}^k - u}{u_{i+p+1}^k - u_{i+1}^k} N_{i+1,p-1}(u) \tag{2.12}$$

and

$$N_{i,0}(u) = \begin{cases} 1 & u_i^k \le u \le u_{i+1}^k \\ 0 & otherwise \end{cases} \wedge \; N_{i,p}(u) = \begin{cases} > 0.0 & u_i^k \le u \le u_{i+p+1}^k \\ = 0.0 & otherwise \end{cases} \tag{2.13}$$

The u_i^k values, *known as the knots*, divide the parametric space into ***uniform*** spans. As a result of the property illustrated by equation 2.13, the curve is divided into a number of segments. The vector containing the values for the curve knots is known as the knot vector,

$\mathbf{U} = \begin{bmatrix} u^k{}_0, & u^k{}_1, & \dots & u^k{}_i, & \dots & u^k{}_m \end{bmatrix}, m = n + p + 2.$

Figure 2.2 illustrates a sample B-Spline curve, and its basis functions.

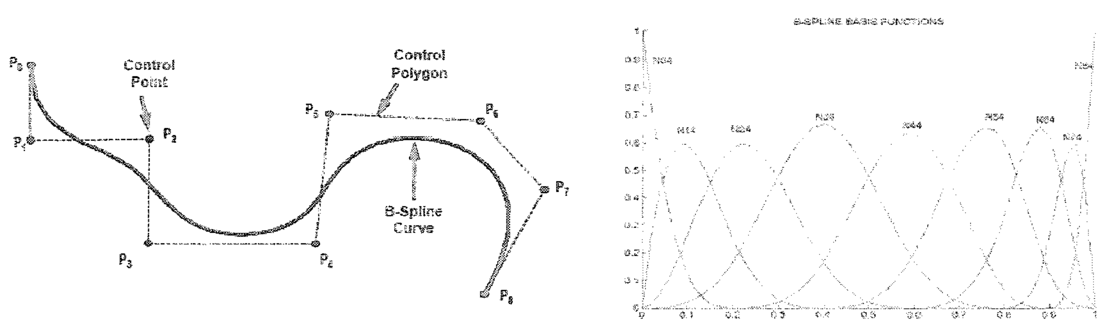


Figure 2.2: Sample B-Spline curve and the corresponding B-Spline basis functions

B-Splines have the following properties of interest:

1. Any given control point $\mathbf{P}_i$ can affect only the region of the curve where $u \in$

$[u_i^k, u_{i+p+1}^k)$. This is known as the *local shape control* property.

2. The curve degree does not restrict the maximum number of control points.

3. Local convex hull property.

4. Curve continuity is inversely proportional to the knot multiplicity.

5. *The end points property*: A curve does not have to pass through the first, and last control points (*unclamped curve*). However, the B-Spline formulation has the flexibility to force the curve to pass through them (*clamped curve*). This is accomplished through raising the multiplicity of the first, and last knots to $p+1$. Figure 2.3 illustrates examples of a clamped, and unclamped B-Spline curves.

6. A B-Spline curve can be converted to a Bézier curve using the *knot removal* algorithm (Piegl and Tiller (1997)).

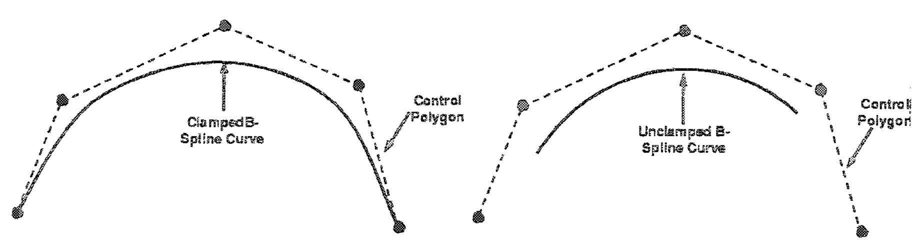


Figure 2.3: Clamped, and un-clamped B-Spline curves

The B-Spline surface representation extends the B-Spline curve representation an extra dimension in the parametric space. The locations of a point on the B-Spline surface can be calculated using equation (2.14).

$$S(u,v) = \begin{bmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{bmatrix} = \sum_{i=0}^{n}\sum_{j=0}^{m} N_{i,k}(u)N_{j,l}(v)\mathbf{P}_{i,j},\ u_{min.} \leq u \leq u_{max.} \wedge v_{min.} \leq v \leq v_{max.} \tag{2.14}$$

$S(u,v)$ is divided into surface patches by the $\mathbf{U}$, and $\mathbf{V}$ knot vectors $\mathbf{U} = \begin{bmatrix} u^k{}_0, & u^k{}_1, & \ldots & u^k{}_i, & \ldots & u^k{}_n \end{bmatrix}$, and $\mathbf{V} = \begin{bmatrix} v^l{}_0, & v^l{}_1, & \ldots & v^l{}_j, & \ldots & v^l{}_m \end{bmatrix}$.

2.2.3.3 Non-Uniform Rational B-Splines (NURBS)

NURBS is a generalized form of B-Splines in which:

1. Knot vectors are not necessarily uniform. Several algorithms for shape parameterizations are reported in the literature (Piegl and Tiller (1997)).
2. Each of the control points is associated with a positive, non-zero value known as the control points *weight.* The weight of a control point emphasizes its effect on the shape of the curve/surface.
3. NURBS utilize *rational basis functions* as their blending functions.

The location of a point on a NURBS surface of degrees p in the u-direction, and q in the v-direction is calculated using equations (2.15), and (2.16 respectively.

$$\mathbf{S}(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} \mathbf{R}_{i,j}(u,v)\mathbf{P}_{i,j} \tag{2.15}$$

$$\mathbf{R}_{i,j}(u,v) = \frac{w_{i,j}\mathbf{N}_{i,p}(u)\mathbf{N}_{j,q}(v)}{\sum_{k=0}^{n}\sum_{l=0}^{m} w_i N_{k,p}(u)N_{l,q}(v)} \tag{2.16}$$

Table 2.3 includes a comparison of Beziér, B-Spline, and NURBS representations of curves and surfaces.

COMPARISON	BÉZIER REP.	B-SPLINES REP.	NURBS REP.
Blending functions	Bernstein polynomials	B-Spline basis functions.	Rational basis functions.
Control points	Number is restricted.	Number is not restricted.	Number is not restricted.
Shape modification	Global.	Local.	Local.
Modeling complex shapes	limited capabilities.	Powerful.	Powerful.
Shape subdivision	Possible using the deCasteljeau algorithm (Piegl and Tiller (1997))	1. Built-in (*using knot vectors.*) 2. Further division possible using knot insertion with *multiplicities*.	1. Built-in (*using knot vectors.*) 2. Further division possible using knot insertion with *multiplicities*.
Degrees of Freedom	1. Control point coords. 2. Shape degree(s).	1. Control point coordinates. 2. Shape degree(s).	1. Control point coordinates. 2. Control point weights. 3. Shape degree(s). 4. Shape *parameterisation*.

Table 2.3: Comparison of parametric curve and surface modeling techniques

2.2.4 Solid Models

Mäntylä (1988) defined solid modeling as:

"The branch of geometric modeling that emphasizes the general applicability of models, and insists on creating **only** *complete physical solid objects."*

There are four main categories of solid models:

1. Decomposition models.

2. Constructive models.

3. Discrete vector models.

4. Boundary models.

2.2.4.1 Decomposition Solid Models

The theme of decomposition models is to describe the physical object using a combination of basic building blocks that are glued together to approximate the shape of that physical object to a certain resolution. Several decomposition techniques have been reported in the literature. Amongst these are:

1. Spacial decomposition: uniform decomposition of the model space using a network of *cubes*. All cubes that intersect with the solid's interior are used to represent it.

2. Cell decomposition: Decompose the model using a grid (*not necessarily uniform*) of cellular element, often with curved surfaces, and glue them to describe the model.

Decomposition models are utilized in limited applications, particularly, finite element analysis, and geographical data bases (Hoffmann (1989)).

2.2.4.2 Constructive Solid Models

An example of this category is the constructive solid geometry (CSG) models. CSG utilizes a number of primitive, well-defined solids to *build* the solid model. Model construction is carried out by applying Boolean operations to the primitive solid models. Regularized set operations are utilized to construct solids that are algebraically closed. CSG trees are the data structures utilized to store the model data in the computer system.

CSG models are efficient to construct models that involve prismatic, and conical features. The capability of CSG to build models is inversely proportional to the model complexity. Furthermore, CSG models are demanding in terms of storage space.

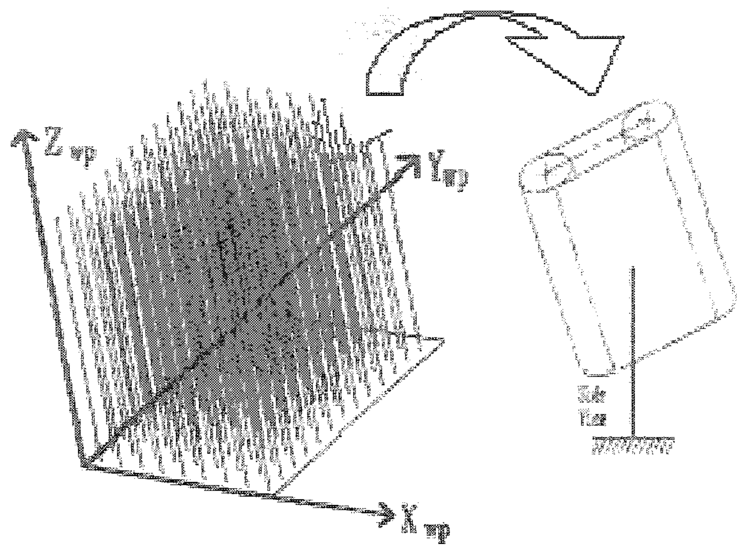


Figure 2.4: Z-buffer modeling (Chappel (1983))

2.2.4.3 Discrete vector (z-buffer) Modeling

The z-buffer method is used for modeling of solids by simplifying the surfaces into a set of parallel vectors with different heights (z-value). The mapping of the surface features to a z-buffer is done by ray firing. A ray is fired toward the surface in the z-direction and the coordinate of the intersection of the ray with the surface is recorded. The crucial step in constructing the z-mapping is the determination of the map resolution or the density of the rays. This density greatly affects the accuracy of the representation and also affects any update process to the solid model. The major drawback of this method (for available algorithms) is that it can only hold on z-value, which makes this method unsuitable for modeling internal features and highly complex surfaces with multiple z intersections.

2.2.4.4 Boundary Models

Boundary models utilize both *topological* and *geometrical* information to define the closed volume of the object's model.

The model geometry defines the types, and dimensions of the elements of which the boundaries of the model are composed. Topology defines the connectivity and associativity of these entities. Both geometry and topology information define the boundaries of the volume of the model.

Boundary representation, as the name implies, utilizes the boundary surfaces of the model to define its closed volume. Information such as the material side of the model are specified. The boundary surfaces are defined using their boundary curves. Curves are defined using the vertices at their extreme ends. The boundary representation of a physical object is considered valid if:

1. Its face set closes.
2. The faces intersect only at common vertices and co-edges.
3. The boundary faces do not intersect themselves.

Boundary representation provides complete definition of the solid model. It has become the basis for most geometric modeling kernels. In modern solid modeling systems, boundary representation involves surface model information. For example, one of the boundaries of a solid could be a NURBS surface. This is thanks to the large body of research (e.g. Varady and Pratt (1984)) that lead to the integration of surface, and solid modeling systems. A comparison of the various solid modeling schemes are included in table 2.4.

COMPARISON	DECOMPOSITION	CONSTRUCTIVE	Z–BUFFER	BOUNDARY REP.
Approach	glue *similar* building blocks.	Use primitive solids	Use fixed–dimension columns	Use object's boundary Information.
Method	Gluing.	Boolean operations.	Trimming operations.	Euler operations.
Representation	Approximate.	Realistic for certain class of objects (*e.g. prismatic*)	Approximate.	Realistic.
Storage	$\propto$ No. of blocks.	$\propto$ No. of primitives and Boolean operations.	$\propto$ dimensions and number of columns.	More compact.
Application	Finite Element Analysis	General purpose.	General purpose.	General purpose.

Table 2.4: Comparison of solid modeling techniques

2.3 Simulation of Multi-axis Milling Process

In industry various types of cutting tools are used. Focusing on the simulation and optimization of CNC milling processes, the tool/workpiece definitions are essential before starting any task.

The tool modeling methods can be classified into analytical, and geometrical models. Geometrical modeling is introduced to deal with simulation of complex tool movements and visualize collision and geometric errors. Geometric simulation is used to evaluate and quantify these errors. There are several methods for geometric modeling both the workpiece and cutting tool. Each method has its level of accuracy complexity.

Analytical modeling, on the other hand, is needed to simulate the physical aspects of the cutting process (forces, deflection, chip movement, etc). Since the sculptured surfaces can be presented in analytical form, the creation of tool models allowed for the creation of mathematical cutting models. The use of each of the modeling techniques (analytical/geometrical) is not exclusive for certain applications, for example mathematical modeling can be used for gouge detection and tool positioning purposes. Also using deformable geometrical models for tool/ workpiece deflection evaluation.

2.3.1 Analytical Representation

In the early stages of the simulation research only the geometric aspects of the process were considered, this simulation was incomplete. If machining constraints are to be satisfied or an improvement in the productivity is required, then the physical aspect of the process must be considered as well.

Simulation of milling forces shows two major phases. The first phase is characterized by using the basic formulae of the un-deformed chip thickness, together with cutting condition parameters including the entry and exit angles and the axial depth of cut, to compute the cutting forces for the case of a flat end mill (El Mounayri et al. (1998)) .

A second phase of milling force simulation is characterized by the introduction of solid modeling to represent complex geometries and extract the geometric data required for NC geometric verification using direct Boolean operations (Voelcker and Hunt (1981)) , alternative approaches like image space Boolean operations and surface based approaches can be found in the work by Abrari et al. (1998), and by Anderson (1987) . The major advance in physical process simulation can be identified from the work by Wang and Wang (1986) and Wang (1988), who used swept volume for simulation of cutter motion. This was followed by a number of other contributions along the same line (Armerago and Deshpande (1989), Melkote and Kendall (1989), Yamazaki et al. (1991)).

Because of the large computational times that are required to calculate the instantaneous chip load over the whole tool path, chip loads are only predicted for one or several revolutions of the tool. This is only the calibration phase of the model, cutting forces are then computed using mechanistic or semi empirical force models(Bailey (2001)).

Mathematical simulation was also used for evaluation of the tool placement

(Abrari et al. (1998), Lee (1998b)) , orientation (Lee and Chang (1997), Lee (1997a)), and collision avoidance that will be discussed in later in the relevant sections.

2.3.1.1 Cutting Edge Modeling

The tool modeling involves two aspects geometrical, and analytical. The model should accurately represent the cutting edge explicitly, and according to the required output the model should simulate the tool motion, and in some cases even the cutter immersion evaluation. A number of different mathematical models has been developed for predicting cutting force in end milling processes, which can be classified into two major categories: analytical and mechanistic methods. The analytical method is to use the cutting parameters (shear stress, coefficient of friction, shear angle) determined from the orthogonal cutting data and then transforms them to obtain the equivalent cutting coefficients of the process (e.g. Sim and Yang (1993), Lee and Altintas (1996)), This method suffers from inaccuracy and tediousness in the measurement of shear angle.

Mechanistic models have been frequently used in industry and academic research

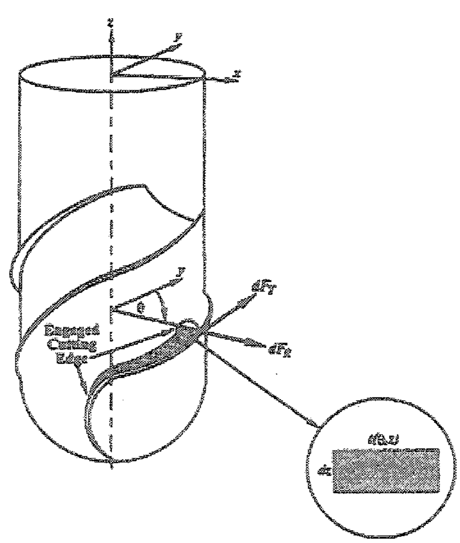


Figure 2.5: Mathematical representation of tool cutting edge (Abrari (1998))

(Bailey (2001), Roth et al. (2003), Gradisek et al. (2004), Abdullahil et al. (2004)). The underlying assumption behind the mechanistic methods is that the cutting forces are proportional to the uncut chip area. The constants of proportionality depending on the cutting conditions, cutting geometry and material properties are determined from an empirical calibration procedure.

Two basic assumptions are made in these models. First, that the kinematics of the milling operation could be modeled by decoupling the motions of the spindle (tool) and table (workpiece). The second assumption is that the mechanics of the end milling process can be modeled by discretizing the tool into thin slices, calculating the cutting force applied to each thin slice of the discretized tool and then summing the differential cutting forces up for all engaged cutting edge elements.

2.3.2 Geometric Representation

The cutter location file (CL) generated by Computer aided manufacturing software packages may involve movements that may cause collisions with fixture elements, generate undercuts, or leave uncut material. The errors in the tool path may produce an out of tolerance part. The purpose of the geometric simulation is to try to avoid such errors. The geometric simulation system must accurately determine the immersion geometry of the cutter as a part is machined; this is usually called the swept volume. The methods of simulation ranges from, a purely mathematical representation to the use of solid modelers to evaluate the tool immersion.

The geometric milling simulation can be classified according to the workpiece modeling and updating techniques into:

1. Surface models.

2. Discrete vector models.

3. Solid instantiation models.

4. Swept volume models.

2.3.2.1 Surface Model

Surface modeling is widely used in defining both the cutting tool and workpiece. Several mathematical and parametric surface representation methods can be found in the literature (Piegl and Tiller (1997)). However the most robust and widely used representation is the (NURBS) surfaces (see section 2.2.3.3).

The surface modeling approach has been used by Lim et al. (1993) and Clayton et al. (2000) to determine the cutter engagement surface in free-form surface machining process. Figure.2.6, shows that the engagement surface can be determined by three boundary curves, namely, the existing surface boundary (LS), the cross-feed boundary (STN) and the feed boundary (LMN). The existing-surface boundary (LS) is the intersection between the existing surface of the workpiece and the envelope of the cutter at the current cutting location. For a ball end mill, the envelope of the cutter can be modeled using a cylindrical surface and a sphere. The existing workpiece surface is usually obtained from the model of the designed surface by some offset. The cross-feed boundary (STN) is the intersection between the cross-feed envelope and the envelope of the cutter at the current cutting location. The cross-feed envelope can be obtained by fitting a surface to the cross-sectional curves along a tool path.

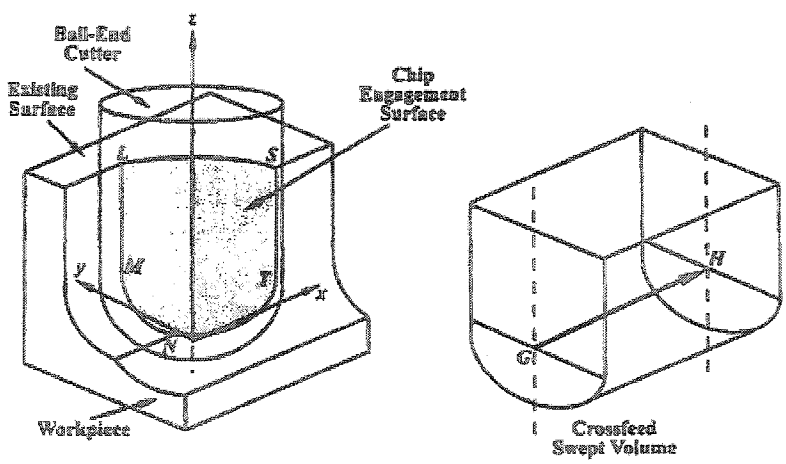


Figure 2.6: Cutter engagement surface (Lim et al. (1993))

2.3.2.2 Discrete Vector (Z–Buffer) Modeling

The z-buffer method is used for modeling of solids by simplifying the surfaces into a set of parallel vectors with different heights (z-value). As shown in figure. 2.4, The modeling of the workpiece in this method can be visualized as a bundle of discrete vectors or grasses (Chappel (1983)), Each point is represented by the height of the surface along Z direction, i.e., Z-map representation (Drysdale et al. (1989)). As the cutter in motion interferes with the workpiece the grass is mowed causing changes in their length. The resolution of mapping is dependent on the surface curvature and the desired tolerance (Kim et al. (1996), Austin et al. (1996)). By using the Z-map, the cutter engagement area can be obtained by comparing the Z-map data of the cutter with that of the machined surface. The workpiece updating is achieved through intersections between the mapped Z-vectors and the tool swept envelope surface. By using the discrete vector approach, the surface/surface intersection calculations can be simplified to the intersection between a line and a surface. But the resolution will greatly affect the output of simulation.

In the work by Fussell et al. (2003), an extended Z-buffer model was used to evaluate the forces during 5-axis machining utilizing a ball end mill (Figure 2.7). This work uses the advantages of the z-buffer method by extending the model so that multiple z values can be stored in each position. The extended z-buffer model is yet to be tested and debugged for various boolean operations before being considerably stable and generally useable.

2.3.2.3 Solid Instantiation

Solid modeling allows complex workpiece geometry and tool path locations to be simulated. The solid model is the most realistic representation of the real world. It can hold volumetric and mechanical properties. It is very useful in performing mechanical and thermal stress evaluation. On the other hand the amount of information included in the solid model makes it computationally complex for geometrical simulation. Most of the geometrical kernels have the ability to transform the solid models into surface

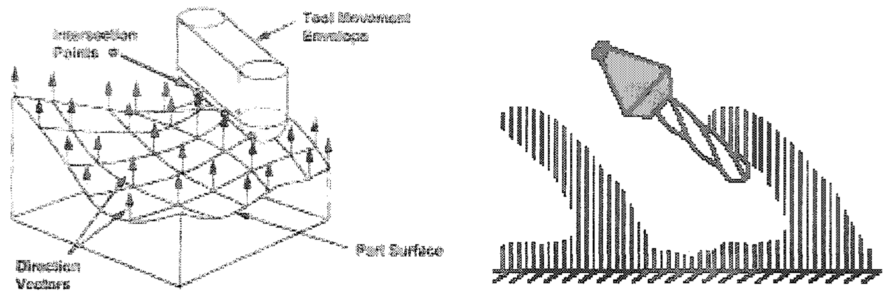


Figure 2.7: Extended z-buffer model (Fussell et al. (2003))

models for faster evaluation by shelling and skinning operations.

2.3.2.4 Swept Volume Evaluation

A swept volume can be modeled much faster and easier using solid modeling than analytical modeling (Spence and Altintas (1994)). Boolean operations are used to determine the intersection of the cutting tool and the workpiece model (solid, surface, faceted, or z-mapped). The work by Spence and Altintas (1994) introduced solid modeling techniques to the simulation of 2.5D milling processes. This work was further extended to 3-axis milling by El Mounayri et al. (1998) , and Imani (1998). In this approach (Figure 2.8), the machined workpiece geometry is updated by subtracting the tool swept volume from the original part body. The cutter engagement surface is then determined by intersecting the cutter body with the machined workpiece. As described in the work By Wang and Wang (1986) , the tool swept volume can be generated by a translational sweeping motion of a solid of revolution. For example for the case of ball end milling, the solid of revolution (i.e., cutter) consists of a cylinder and a semi-sphere. In general, ten faces are required for the construction of a valid 3-axis cutter swept volume, as shown in Figure.2.9. The boundary of swept volume consists of two categories of faces. Analytical faces (Faces 5 to 10) represent portions of the cutter at initial and final positions. The second category of faces (Faces 1 to 4) are tool path dependent and can be constructed by using sweeping techniques. A

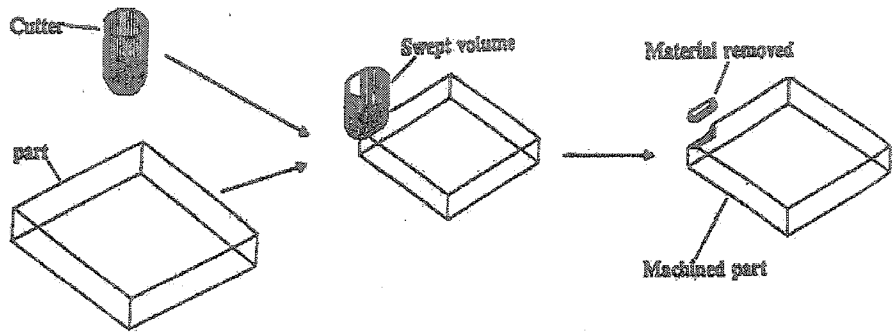


Figure 2.8: Solid modeling based milling simulation (El Mounayri et al. (1998))

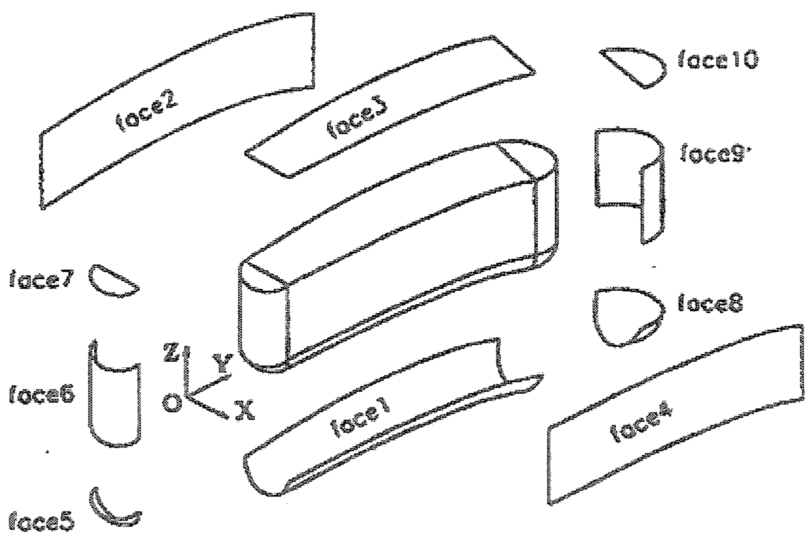


Figure 2.9: Swept volume surfaces construction (Imani (1998))

more generalized translational swept volume can be found in (Piegl (1991)). A major drawback of this method is the self intersection of the swept volume during multi-axis machining(Tounsi et al. (2002)). This can be eliminated by sectioning of the tool path into small segments and evaluating each part in separate boolean operations. This will lead to more complications in terms of the number and the shapes of generated surfaces and may increase the evaluation time significantly.

Table 2.5 summarizes the characteristics of different combinations of milling simulation systems.

Tool / part	Description	Advantages	Disadvantages
Solid / solid	1. Generates copy of the tool at each path position along the tool path. 2. Perform subtraction operation to generate the updated part	1. The most realistic simulation to cutting path. 2. Successful in intersecting tool path evaluations	1. Computationally intensive and slow. 2. Requires high sampling rate to accurately simulate the substitute geometry.
Swept volume/solid	Swept volume/solid The tool profile is swept along the tool path and solid subtraction operation is done to the solid workpiece model to generate the updated part.	1. The generation of the updated geometry is faster than solid intersection 2. Suitable for smooth tool path transitions	1.Fails to deal with the sharp and intersected tool path 2. With complex and long tool path the construction of swept volume became computationally intensive and slow.
Swept volume /z−buffer	1. Construct z-buffer model of the semi-finished part by ray firing techniques 2. Generate cutting path swept volume.Intersect the swept volume with the part z-buffer model. 3.Trim the z-buffer model after intersection to create the updated workpiece	1. Faster than solid intersection methods 2.Numerically stable and easier to update	1. Requires very dense grid to maintain reasonable accuracy of the workpiece model, which increase the computational time. 2.Fails to deal with the sharp and intersected tool path
Solid /z−buffer	1. Generates copy of the tool at each path position along the tool path. 2. Intersect the tool with the z-buffer to update the workpiece model.	1. More accurate than the swept volume/z−buffer method. 2. It can handle sharp and complex cutter path simulation.	Requires very dense grid to maintain reasonable accuracy of the workpiece model, which increase the computational time.

Table 2.5: Tool/workpiece simulation overview

2.4 Tool path planning strategies

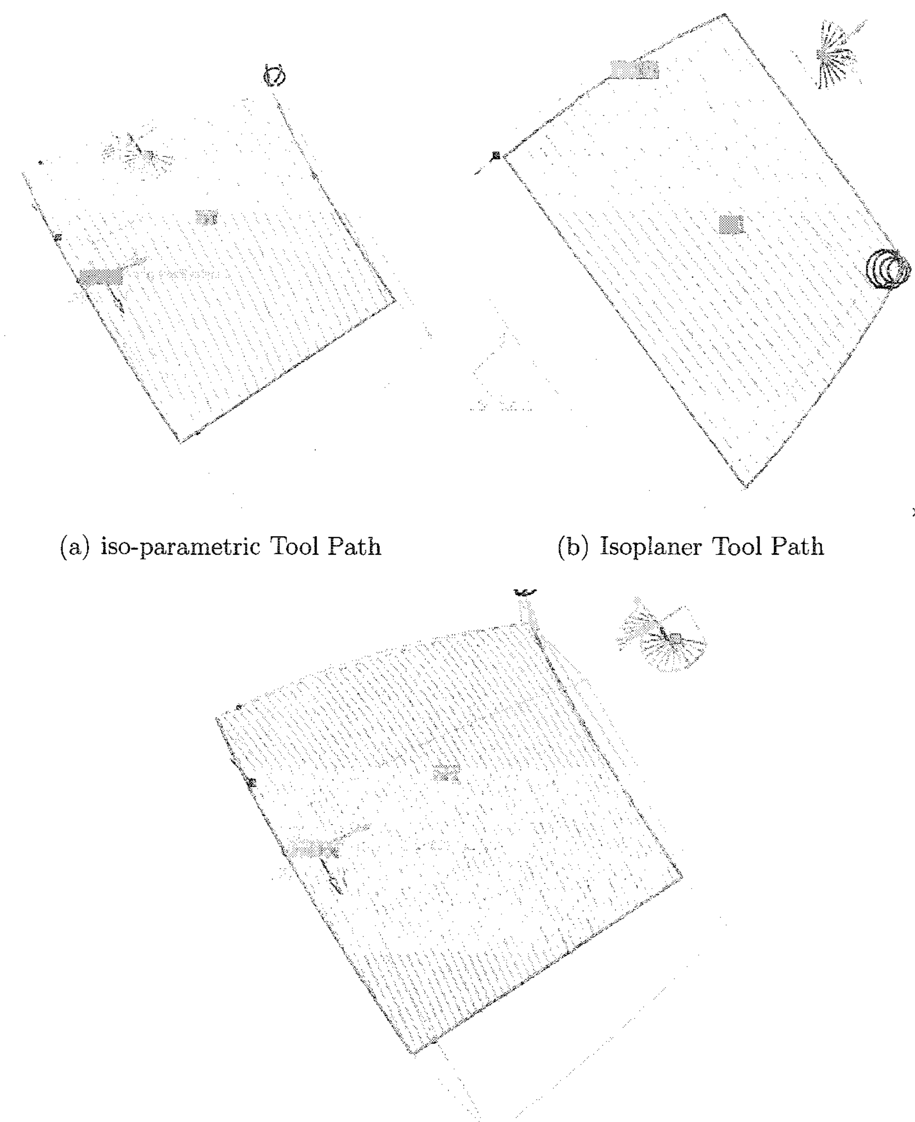
The tool path planning methods can generally be classified into four techniques; iso-parametric, isoplanar, constant cusp and machining strip evaluation.

In the isoplanar tool path method as shown in figure 2.10(b), parallel planes are created to generate intersection curves with the part. These curves are used to guide the cutting tool's movement. This method while maintaining a uniform path, does not take into consideration the topology of the machined part.

The iso-parametric tool path planning method, shown in figure 2.10(a), considers the fact that the part surface is usually represented as a free-form surface. This surface is mapped into the parametric $(u,\ v)$ domain. By keeping one of the parameters constant, the iso-parametric curves generated, are used as a guide for the tool path. This method creates non-uniform path segments, but less surface errors. The generated tool path is often denser in some places than others depending on the number of knots defining the surface patch.

A constant cusp height as shown in figure 2.10(c), relies on the inspection of adjacent path curves to control the cusp height at the start point of each path. This method is widely adopted in 3-axis machining with a ball end mill, since the cusp height is determined by the intersection of two circles (see Figure 2.11).

The evaluation of machining strip method is used in five-axis machining, with cutting tools other than a ball end mill. It evaluates the imprints of the tool on the workpiece surface and. constructs a machining strip by connecting the tool imprint lines. Then the distance between each two adjacent strips is to be evaluated and controlled to minimize the surface errors.



(a) iso-parametric Tool Path (b) Isoplaner Tool Path

(c) Constant Cusp Tool Path

Figure 2.10: Common tool path planning methods

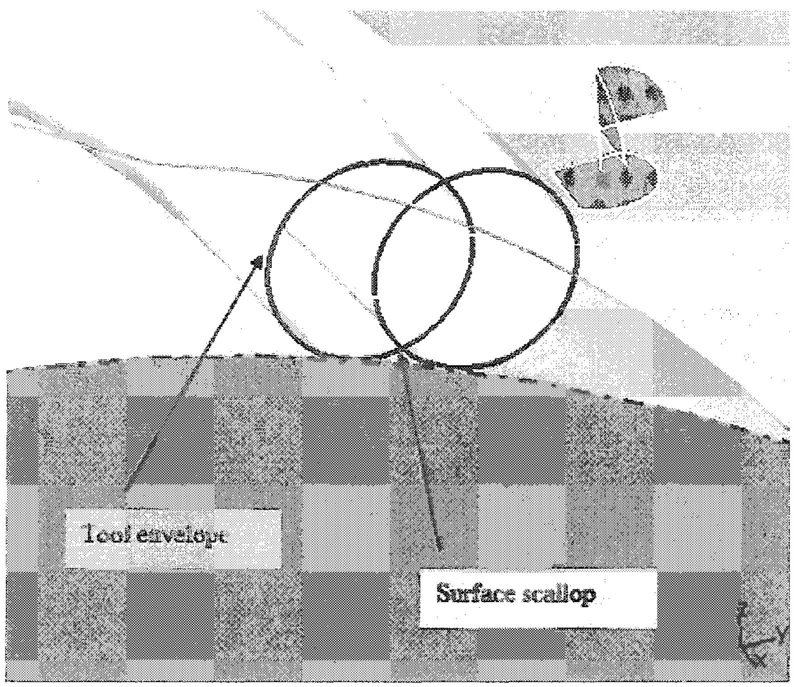


Figure 2.11: Cusp height calculation in ball end milling

2.4.1 Iso-parametric Evaluation

In iso-parametric path planning (Figure 2.10(a)) one of the surface parameters (u or v) is kept constant while generating the tool path (e.g. Pi et al. (1998)). Chen and Ravani (1987) and Kim and Kim (1995), used an offset curves approach while Li and Jerard (1994) used evaluation points to investigate the surface.

Since the u or v lines in the parametric domains are always, mapped to nonlinear curves in the Cartesian domain, unequal stepover in the tool path is created, resulting in difficulty in assessing the cusp height. Several attempts have been made to control the cusp height. The work by Elber and Cohen (1994) suggested adding or removing intermediate iso-parametric curves to keep the cusp height within an acceptable range. This method results in denser tool path in some surface regions.

2.4.2 Isoplanar Evaluation

In isoplanar path planning illustrated in figure 2.10(b), equally spaced intersection planes are used to generate the intersection curves utilized to create the tool path (Huang and Ovliver (1992)). The distribution of these planes is parameterized in the Cartesian space regardless of the part surface topology.

2.4.3 Constant Cusp Evaluation

In constant cusp path planning (Figure 2.10(c)) the cusp height is maintained constant by checking the adjacent tool position generated by the tool. This operation is suitable mainly for ball end milling operations. Several constant cusp height work can be found in (Pi et al. (1998), Lee (1998c), Feng and Li (2000)).

2.4.4 Machining Strip Evaluation

The research in this method was started by Lee (1998b) for 5 axis machining of die and molds using an iso-parametric tool path, the technique analyzes the machining strip width, its separation and allows for strip width variation throughout the change of inclination angle of the cutting tool. In the work by Lee (1998a) the machining strip width evaluation method is used to generate the tool path by comparing the distance between two adjacent strips, and tries to minimize the spacing between them.

2.4.5 Curvature Based Evaluation

In these methods, three sampling techniques are mostly used; equiparametric, surface batch size, and surface batch mean curvature. The applications of these methods are found in the inspection planning literature, but because they are related to the surface representations they can be used in cutting path planning as well.

2.4.5.1 Equiparametric Evaluation

This method distributes the sampling points equally in the parametric u-v domain. This method is simple to implement but insensitive to sharp curvature changes.

2.4.5.2 Surface Batch Size Evaluation

In this method (Figure 2.12), the surface is divided into batches and ranked according to each batch size. The higher the rank the larger the number of sampling points is allocated to it. This method is more involved with the surface topology than equiparametric sampling, but it carries the risk of ignoring some small-sized patches with sharp curvature changes.

2.4.5.3 Surface Batch Mean Curvature Evaluation

In this method the surface is divided into batches as in the previous method, then for each parametric point on the surface the curvature is calculated and the patches are then ranked by the value of their curvature (higher rank for higher curvatures). The sampling points are then distributed according to the batch rank

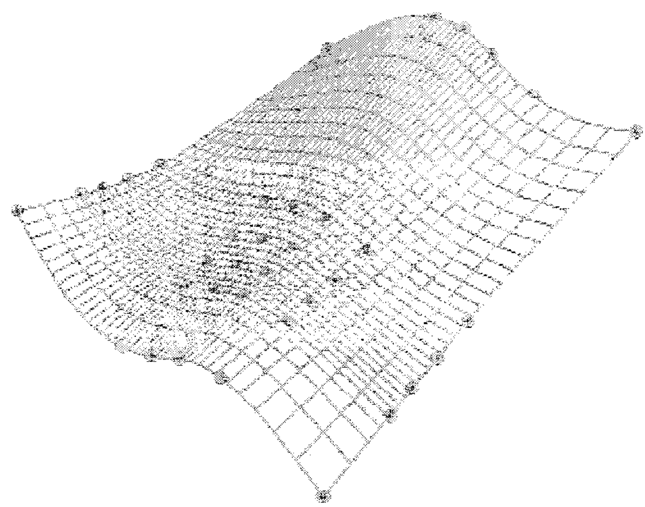


Figure 2.12: Surface batch size-based evaluation (ElKott et al. (2002))

2.5 Modeling of Cutting Tools

The simulation, and optimization processes require the definition of mathematical and/or geometrical models for both tool and part. Different cutting tool applications require the definition of various tool models. To create robust simulation processes, generic tool models are developed. These models, both mathematical and geometric, are created to define the key features in the most frequently used cutting tools. The degree of the sophistication of the cutting tool model is dictated by the amount of data needed to perform the task in hand. In some cases the simulation of the cutting edge is the most important, while in other cases the simulation of the tool's outer profile is sufficient to complete the simulation. Hybrid models are sometimes used to integrate different cutting process simulation and optimization modules.

2.5.1 Geometric Models

In Geometrical methods, the envelope or outer geometry of the cutter is used in generating NC tool paths on CAD/CAM systems (Figure 2.13). Moreover, the envelope of the cutter is used in identifying the intersection of the cutter and workpiece geometry, which is required for simulating the material removal process and in dynamically updating the blank geometry for graphical NC tool path verification (Spence and Altintas (1994),Leu et al. (1997), Gu et al. (1997)).

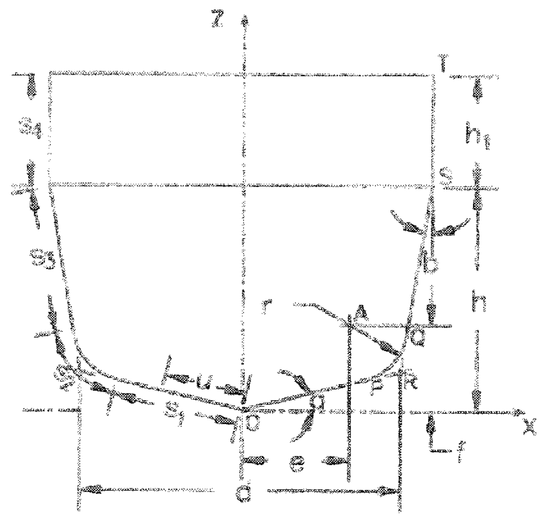


Figure 2.13: Generic tool model (Karunakaran et al. (2000))

2.5.2 Analytical Models

Several researchers used analytical modeling to describe the tool. In the work by Engin and Altintas (2001), (Figure2.14) the generalized tool is parameterized by seven geometric parameters. These seven geometric parameters are independent of each other, but the relations and limits are constrained to create mathematically realizable shapes. Other research by Lee (1998b), used different number of parameters and used different models for different tool categories. In Figure 2.15 the tool surface Ψ_{tool} is described in the tool coordinate system (X_T, Y_T, Z_t) in equations (2.17), (2.18), and (2.19).

Ball EndMill

$$\Psi_{sphere}(\theta,\phi,\beta)_T = \begin{pmatrix} R_2 \sin\theta \sin\phi \\ R_2(1-\cos\theta)\cos\phi \\ R_2 \cos\theta \sin\phi \end{pmatrix} \tag{2.17}$$

Flat EndMill

$$\Psi_{Flat}(\theta,\phi,\beta)_T = \begin{pmatrix} \alpha R_1 \sin\theta \\ \beta H_{flank} \\ \alpha R_1 \cos\theta \end{pmatrix} \tag{2.18}$$

Filleted EndMill

$$\Psi_{Fillet}(\theta,\phi,\beta)_T = \begin{pmatrix} (\alpha R_1 + R_2 \sin\phi)\sin\theta \\ R_2(1-\cos\phi) + \beta H_{flank} \\ (\alpha R_1 + R_2 \sin\phi)\cos\theta \end{pmatrix} \tag{2.19}$$

From figure 2.15, $\theta \in [0, 2\pi]$, $\phi \in [0, \pi/2]$, and α, β are shape coefficients This allow for different tool geometries to be used in the evaluation procedures.

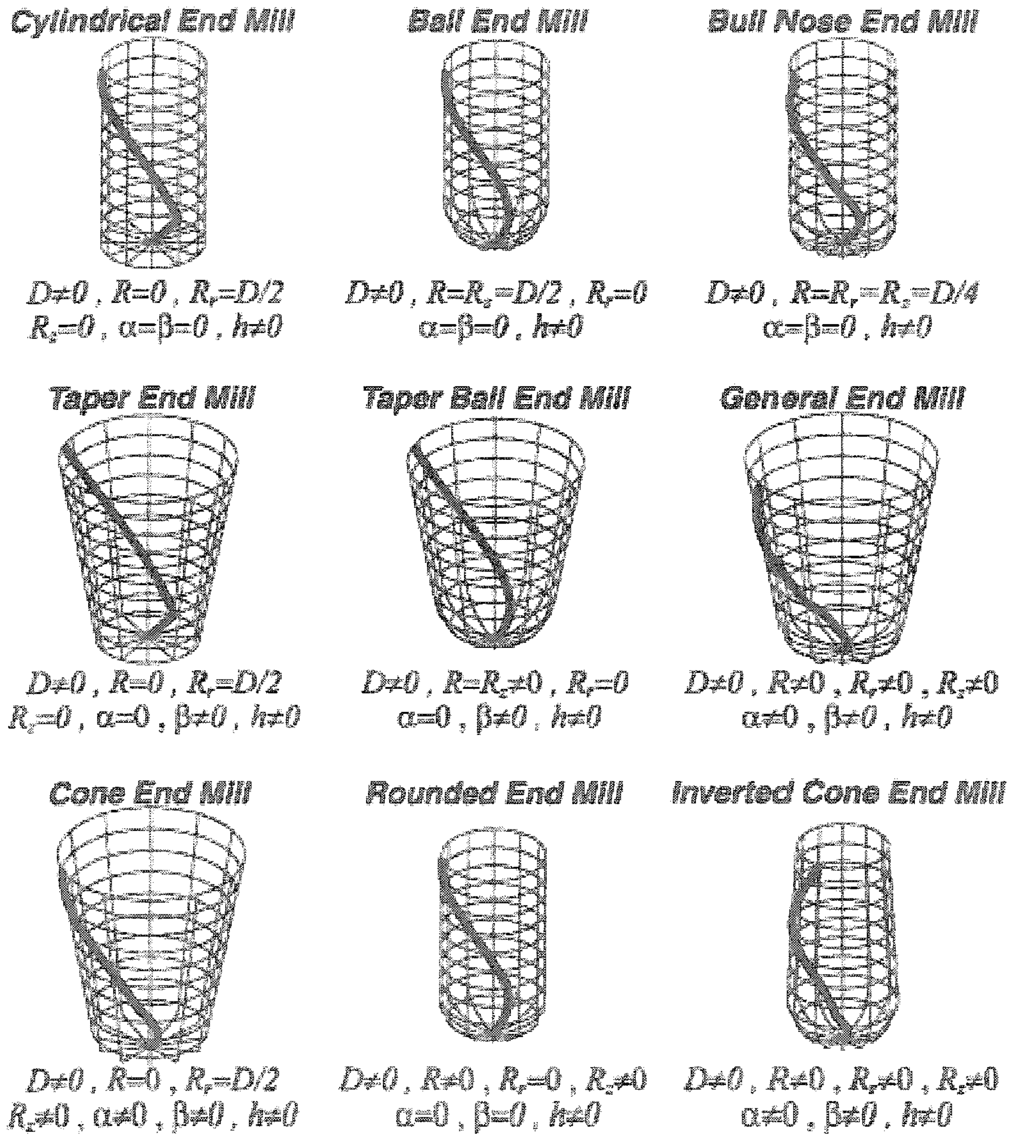


Figure 2.14: Cutting edge representation (Engin and Altintas (2001))

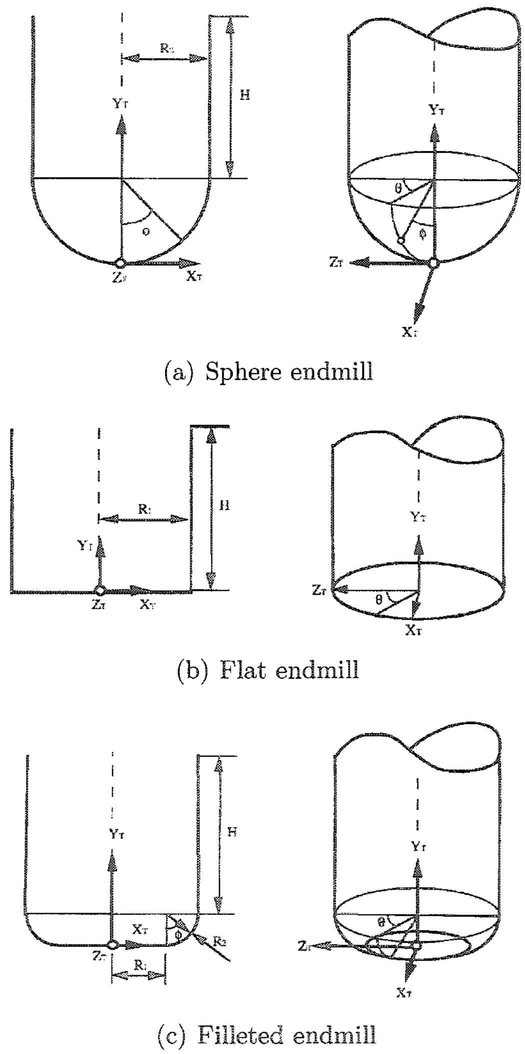


Figure 2.15: Mathematical models Of different endmill cutters Lee (1998b)

2.6 Tool Posture and Tool Positioning

Five-axes milling allows machining of free form surfaces with cylindrical or filleted end mill cutters instead of ball nose cutter. This drastically reduces the machining time versus ball end milling operations. On the other hand errors in programming tool path for complex surface results in gouging and undercuts. This has prompted the development of graphical display indicating where errors have occurred. Errors in tool path generation were typically determined using dry cut path checking and graphic visualization techniques (Marshall and Griffiths (1994)).

In multi-axis milling the tool orientation always varies to ensure contact between the tool and the part. This is essential to avoid the creation of surface errors. Each tool posture is described by its tool tip coordinates (x, y, z) and tool orientation (a, b, c), expressed in the part coordinate system in most cases. The tool positioning problem can be classified based on the cutting tool type into either ball nosed end mill, or another standard cutter type. The reason for this classification is that most of the positioning methods use the effective tool profile, or tool silhouette, to evaluate the tool orientation. The ball nosed end mill's effective profile is a circle regardless of its orientation. Therefore, for the ball nosed end mill, the positioning problem is simplified into accessibility and collision avoidance problems.

On the other hand the tool positioning problem for other types of milling cutters is dependent on the effective cutting profile and related to the workpiece topology. The tool positioning problem cannot be discussed apart from the tool selection problem, since the main reason for cutting process optimization is to select the maximum tool size that will match the surface topology and result in minimum deviation from the design geometry.

In five-axes milling, intensive user interaction is needed for using CAD/CAM software to generate NC part programs for sculptured surface machining (Vickers et al. (1990), Kim and Chu (1994)). Automation of planning tasks in necessary to take full advantages of 5-axis machining.

Traditionally, the tool orientation is defined by a lead angle and tilt angle. The lead angle is defined in a plane parallel to the feed direction, while the tilt angle is defined perpendicular to it. Most CAM systems for multi-axes milling only allow the user to define a constant lead and/or tilt angle. Some software packages apply a variable (optimized) lead angle, which is optimized based on the local curvature of the part surface. Several approaches for solving the cutter placement problem were developed. Some methods which calculate the offset surface and plans the center of the cutter moving on the offset surface (e.g. Chen and Ravani (1987), Kim and Kim (1995)). These methods are for a ball-end cutter and cannot be easily extended to a non-ball-end mill cutter.

In other research (e.g. Oliver and Goodman (1991), Li and Jerard (1994), Jensen et al. (2002b)), a set of polygons is generated from the model of a surface and used to generate the cutter location data. Because the generation of cutter location is based on polygons, the local differential geometric features of the designed surface are largely approximated.

Another method discussed in Jensen and Anderson (1992) and extended by Jensen et al. (1995) is based upon effective cutter radius calculations. It was applied to determine the location of a flat-end cutter in 5-axis machining,. They used the local differential geometrical features of the machined surface to determine the cutter location by matching the curvature of the cutter silhouette to the curvature of the surface. The silhouette is the projection of the cutter bottom in a plane perpendicular to the cutting direction. This method was extended to compensate for gouging and collision in the work by Jensen and Anderson (1992). Another method based on cutter projection considered the radius of curvature of the cutter as a function of both inclination and yaw angles (Deng et al. (1996), Lee (1997b), Rao et al. (2000)).

A mathematical approach by Lee (1998b) was proposed to place (position) filleted end mills (Figure 2.16), which intersect the mathematical model of the tool workpiece to find the suitable tilt (yaw) and inclination angles.

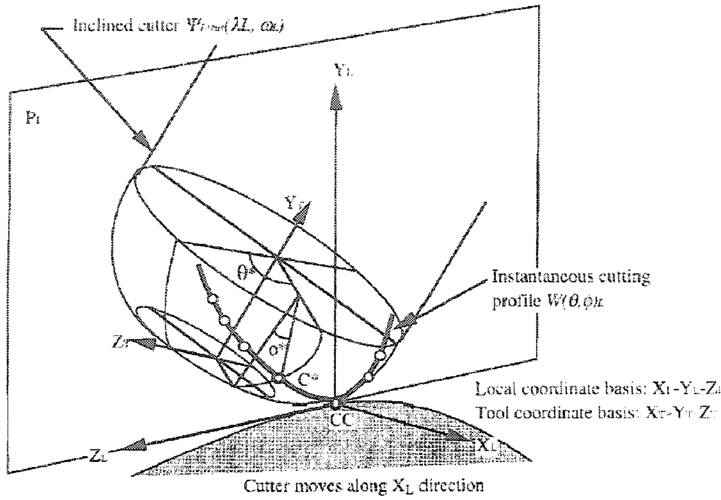


Figure 2.16: Fillet endmill cutter positioning (Lee (1998b))

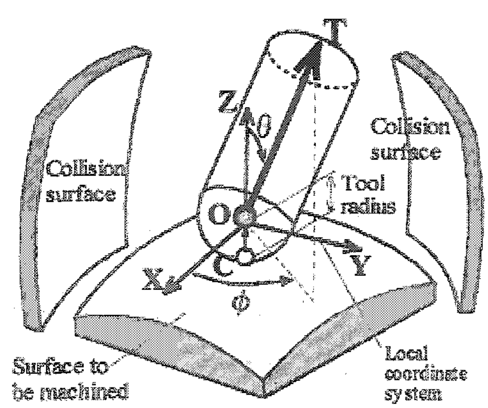
In the method by Lee (1998c), he proposed the use of non iso-parametric path that integrate with his previously developed machining strip evaluation concept. The machining strip was defined as the machined region that lies within the given surface tolerance.

2.7 Gouge and Collision Detection and Avoidance

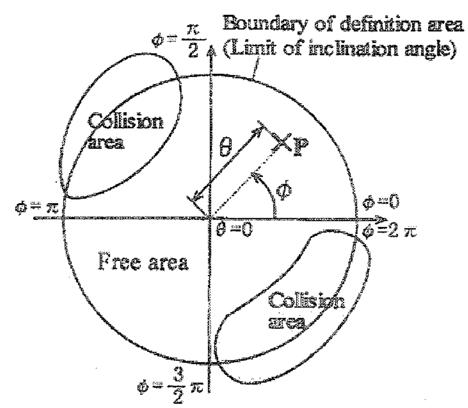
The tool positioning criteria discussed in the previous section would not be complete without collision and gouge detection. As shown in figure 6.1(a) and 6.1(b), although the tool positioning with respect to the workpiece surface (contact point) was performed correctly, the tool size and orientation may cause the rear part of the tool to cut into the finished part and the tool shank to hit the part while trying to reach the cutting area.

Several methods were developed to avoid collision and gouging. For example the calculation of the nearest distance between tool center line and the workpiece and comparing this value with the tool geometry to detect gouging and collision. Other techniques perform intersection checks mathematically and compensate for its value by varying the tool inclination. The accuracy and speed of these methods are greatly affected by the modeling and tool path simulation methods. For example some techniques, for the purpose of increasing the calculation speed, approximate the workpiece by a polyhedron or faceted surface. This simplifies the intersection check from cylinder/ surface or solid intersection evaluation to cylinder/ plane evaluation. The choice of the degree of simplification is to be characterized by the degree of accuracy required in the the final product.

The work done by Morishige et al. (1997) and Jun et al. (2003), developed an approach for collision avoidance in ball end mill 5-axis machining by the simulation of the obstacles encountered by the cutter. This 3D approach (c−space approach) shows the relationship between tool positions, posture and the collision areas. This approach is used in the manufacturing of complicated gas turbine impellers. Figure 2.17(a) illustrates the definition of a cutter location in real space. According to the tool orientation and the point of contact of the tool with the workpiece surface 2-D C-space is created by the projection of the collision surfaces in the direction of the tool orientation (Figure 2.17(b)).



(a) Definition of tool posture in real space



(b) Projection of C space

Figure 2.17: C space definition (Morishige et al. (1997))

In the work by Balasubramaniam et al. (2002), a method for collision detection and tool accessibility envelope assessment is proposed (Figure 2.18). This method relies on haptic surface generation by manually positioning the tool in order to reach the requires surface and record all the possible orientation, though this method generates one hundred percent collision free tool path, it is unrealistic and time consuming to follow each path point manually to learn all the tool positioning possibilities.

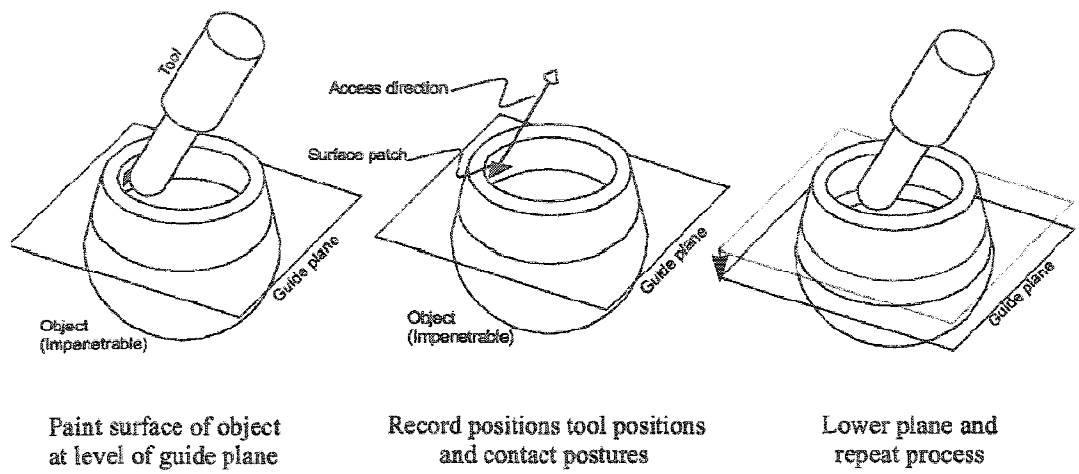


Figure 2.18: Definition of tool posture in real space (Balasubramaniam et al. (2002))

2.8 Flank Milling of Ruled Surfaces

The majority of the work in five-axis milling is focused on point milling, while there are more productive methods to achieve some tasks.

Flank milling can provide higher productivity than point milling for a certain class of surfaces. Most blades, integrated turbines, and fan surfaces fall into the category of flank milling-enabled applications. This method utilizes a greater area of the tool surface, the tool shaft or side surface, resulting in a higher metal removal rate per machining pass. The major limitation of this technique is that it is exclusive for certain classes of surfaces such as ruled surfaces. Other surface types maybe very difficult and even sometimes impossible to utilize the flank milling process.

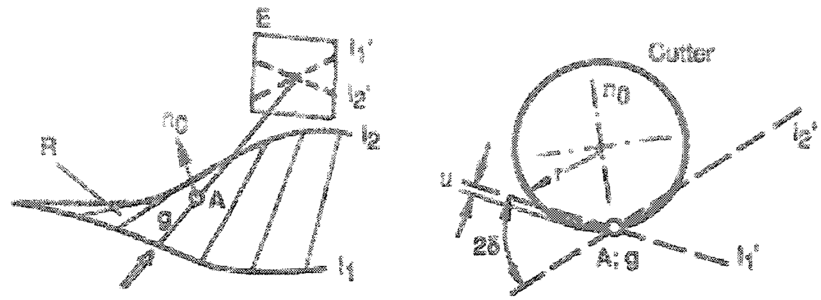
In the flank milling method, for full contact between the tool and the part surface, only ruled surfaces created by straight ruling lines can qualify for this method. This class of surfaces is called developable ruled surfaces.

One of the first research studies done in this area was by Stute and Storr (1979), they dealt with straight ruled surfaces that have directrices perpendicular to the ruling curves. The tool is positioned tangent to the rule line.

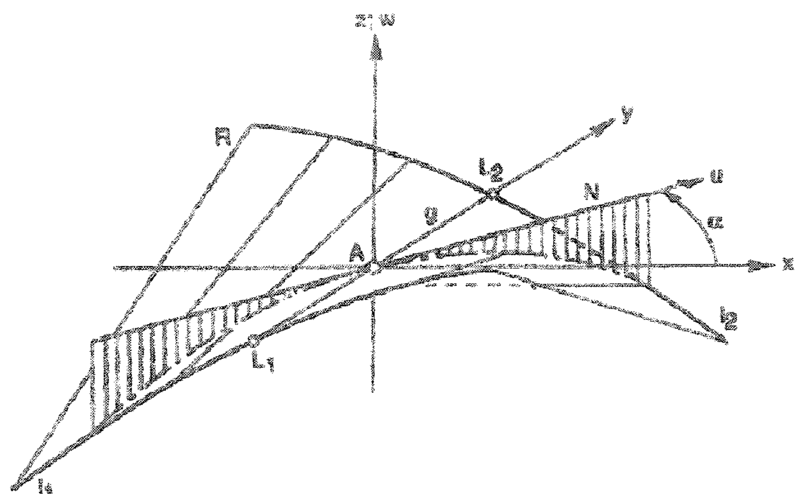
In the work by Rehsteiner and Renker (1993) oriented toward developable surfaces and Marciniak (1991) for non−developable surfaces, the problem of positioning the tool relative to the twisted ruled surface was addressed, the method suggests that the tool be placed parallel to the rule-line, and in contact with the surface at its middle point on the rule line (Figure 2.19). The method while avoiding collision, suffers from large deviations from the designed surface. Another study utilizing the same positioning principles was developed by Bohez et al. (1997), they interactively redesigned the surface to reduce the error within the design parameters while for further reduction of the over-cuts multiple tool passes are considered.

In another attempt to reduce the error, Liu (1995) suggested dividing the error to over-cut and undercut. The tool is then positioned by calculating the contact points at points located at quarter and three quarter the length of the ruled lines. This method managed to reduce the amount of the error produced by the previous positioning techniques.

Redonnet et al. (1998) proposed a strategy of placing the tool tangent to two rule-lines and two boundary curves. This method is mathematically intensive and seek the exact solution of the positioning problem, thus it is very slow. Other work by the same research group (Monies et al. (2000)), suggested the use of a conical mill instead of a cylindrical cutter. The positioning is done by the same method, and then an algorithm is applied for the rotation of the tool and to analyze its error. The study approximate the surface by quadratic equations, which add approximation error to the other calculation errors. The research was then continued (Monies et al. (2002)), by



(a) Twisted ruled surface contact points

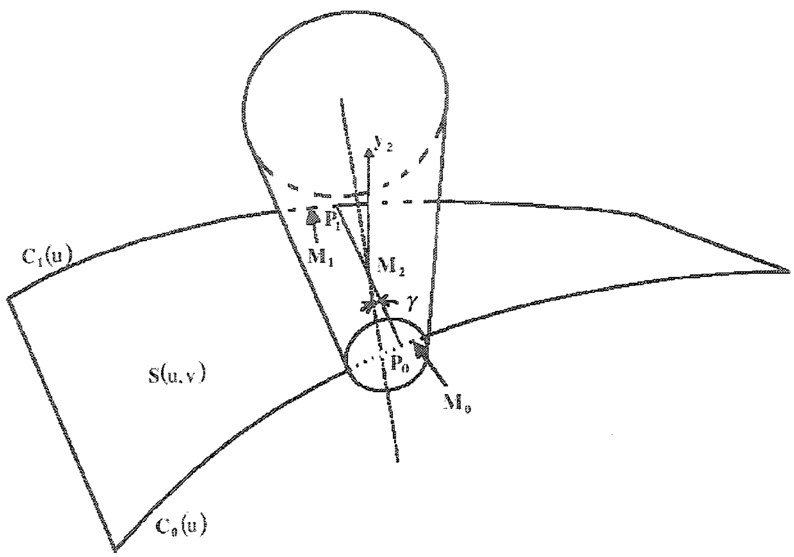


(b) The uncut region

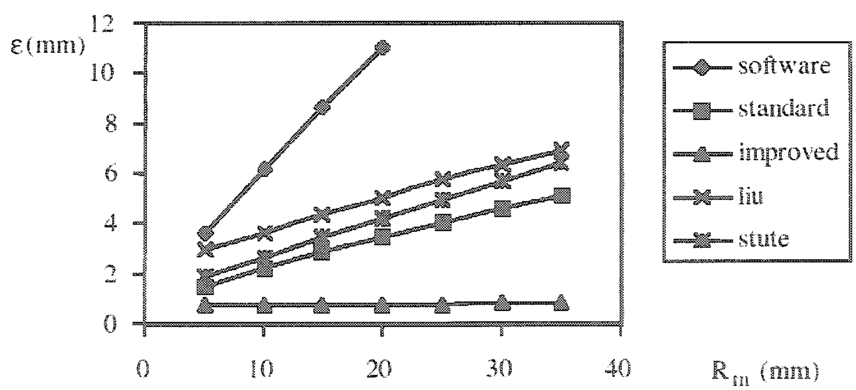
Figure 2.19: Flank milling errors in twisted ruled surface (Rehsteiner and Renker (1993))

implementing an algorithm for tool selection based on the minimum surface curvature and the given tolerances. As seen in figure 2.20(a), the method relies on keeping the tool in contact with the surface in three points M_2 on the rule line, M_0 and M_1 on the two directrices. A comparison of the error produced by this method versus the other previously discussed methods, can be seen in figure 2.20(b). It can be inferred from the figure that this method dramatically reduced the error for different cutter radii (R_m).

Another research group adapted the concept of keeping the tool tangent to the directrices throughout the tool path(Bedi et al. (2003)), this method while simple produces over-cuts in the middle. In an attempt to reduce the error, further research was carried by Menzel et al. (2004), in which the tool positions from the first algorithm are pushed inwards along the rule line until the error is minimized, another step is then taken by sliding the tool contact point sideways to minimize this new error,this will result in undercuts in the areas outside the contact points. A decision should be taken to choose the balance between over-cuts and undercuts(Figure 2.21).

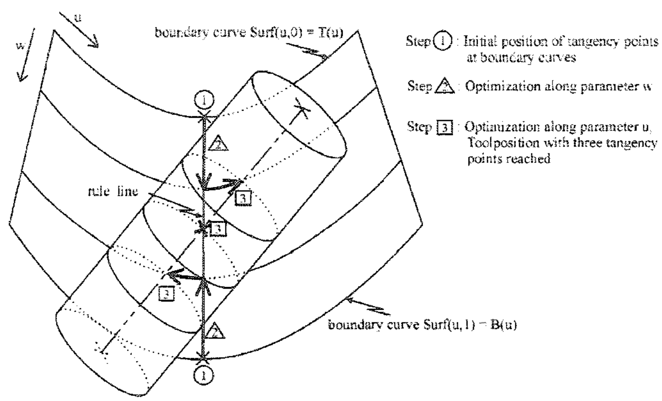


(a) Improved positioning

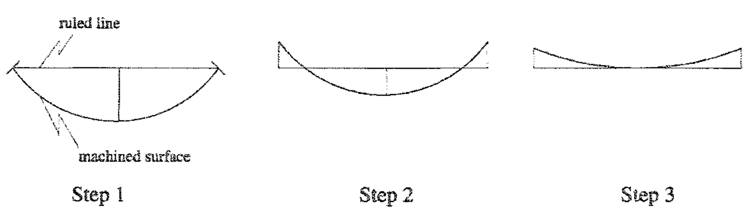


(b) Error (ε) results

Figure 2.20: Improved flank milling using conical cutter(Monies et al. (2002))



(a) Search Algorithm



(b) Surface Deviations

Figure 2.21: Flank milling deviation reduction Algorithm by (Menzel et al. (2004))

2.9 Error Reduction

The accuracy of the machined part is very critical to any manufacturing system. Various sources of machining errors contribute to the deviation from the designed specifications. These sources include thermal expansion, tool deflection, feed drive lags, fixture dynamics and CAM system errors. Thus the assessment of these errors is very important to close the feedback loop in a manufacturing system and find a means to reduce the errors (Capello and Semeraro (2001)).

After the creation of the tool path, several methods are used to update the workpiece geometry to reflect the effect of applying the CL-Data to the stock/ semi-finished part.

These methods range from mathematically representing the tool/workpiece system to purely geometric modeling. In geometric modeling several combinations of tool/ workpiece modeling techniques are found in the literature (solid/ solid, solid/z-buffer, swept volume/solid, swept volume/z-buffer) each combination has its advantages and disadvantages. From table 2.5, it can be seen that the choice of the optimal method is a matter of balance between time and accuracy.

Tool path approximation is considered one of the machined surface error sources. In most of the CAD systems the tool path is approximated by line segments (Ho et al. (2003)).

Ramesh et al. (2000), reviewed the sources of the geometrical errors including the effect of cutting forces. The work by Lee and Chang (1996) proposes an algorithm for finding errors based on an analytical model of the tool and workpiece. The intersection of two adjacent tool shapes is used to asses the cusp height and relate its value to a given tolerance. This method is suitable for path planning rather than error assessment and can be used to minimize the errors during the planning phase.

CHAPTER 3
CURVATURE BASED TOOL PATH PLANNING

3.1 Evaluation of Workpiece Data and Sampling Strategy

In industry various types of cutting tools are used. Focusing on the simulation and optimization of CNC milling processes, the tool/workpiece definitions are essential before starting any task. The workpiece modeling methods can be classified into mathematical and geometrical models. After the retrieval of the workpiece data, tool path evaluation and planning can be carried out. The tool path planning methods can generally be classified into four main techniques; iso-parametric, isoplanar, constant cusp and machining strip evaluation. This work used the iso-parametric surface evaluation method as a starting point. The reason was that the iso-parametric parameterizations is the most common form for free−form surface representation and that this method used the surface topology to evaluate the tool path points. The improvement in such a method will lead to faster conversion toward the desired tool path smoothness and accuracy.

In the next sections the workpiece interrogation and error evaluation and compensation algorithms used in this work will be discussed. The framework for this algorithm is illustrated in figure 3.1. More detailed flowcharts can be found in appendix A.1.

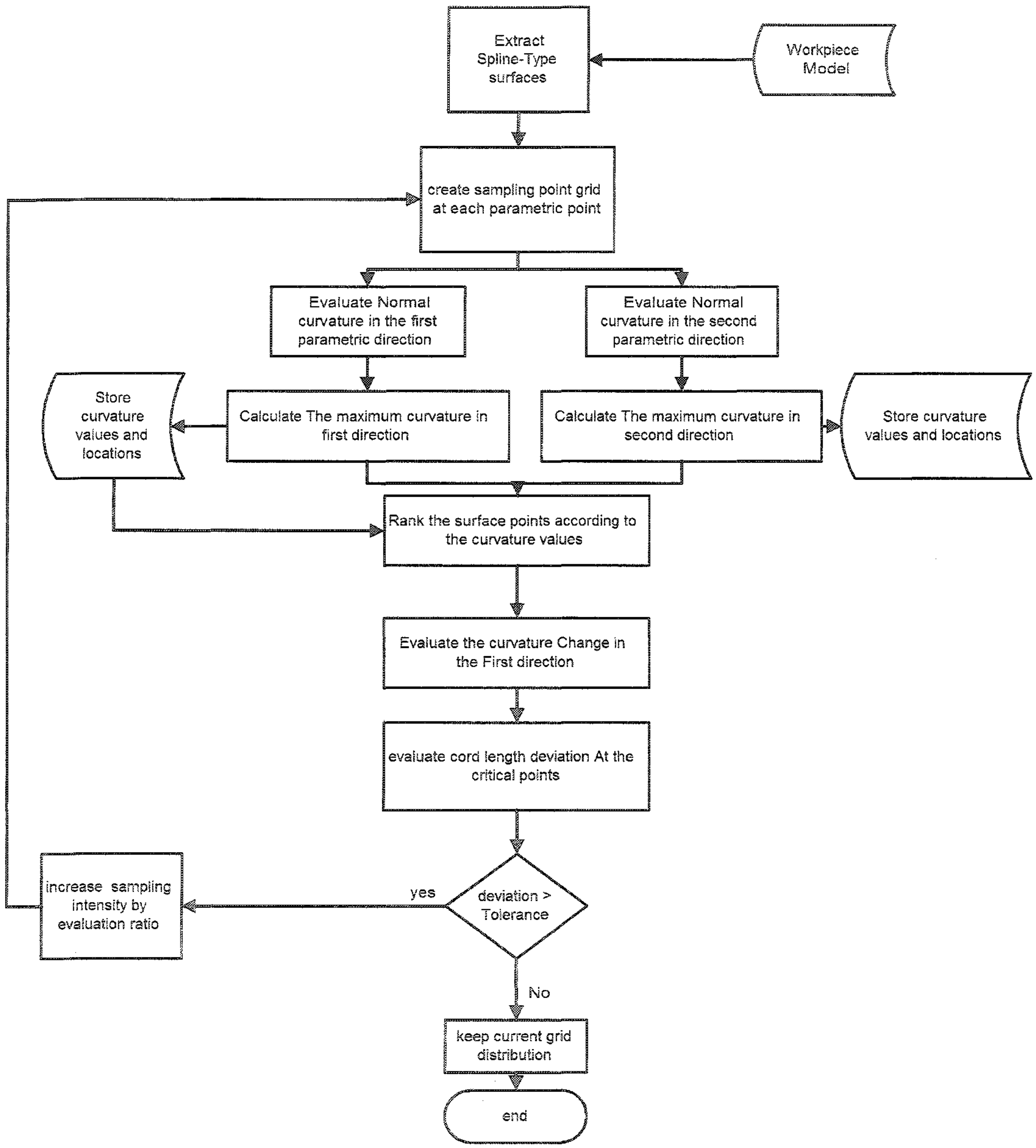


Figure 3.1: Workpiece topology evaluation

3.2 Interrogation of Free–form Surfaces

The goal of the optimization for five axis milling of a free–form surface process is to get the optimum tool movement that will produce the desired geometry within the given tolerances of the part. To achieve such a goal the first step is to carefully interrogate the surface properties. The starting point is to retrieve the workpiece model. Various geometrical file storage formats are available in the CAD/CAM environment. Depending on the CAD system used the output format may vary. Several commercial CAD software packages can convert between the different file formats. The geometrical kernel that was used for the development of this work is ACIS version 6.3 by Dassult Systèmes. The native file format for this kernel is SAT files. The SAT files are text files describing the model topological relationships and their geometrical entities. The Kernel also can deal with IGES(Initial Graphics Exchange Specification) files and CATIA part files.

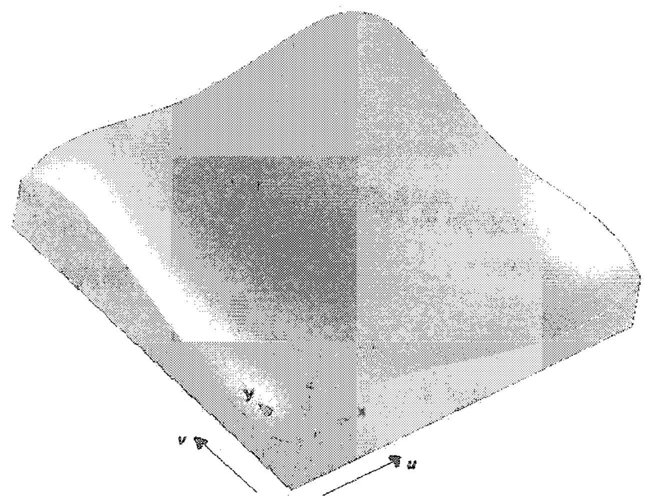
On retrieving the model file the system inspects the solid model and searchs for the free–form surfaces. The search algorithm inspects each surface of the model searching for **quadri-lateral** *spline* surfaces, the category under which the NURBS surface is defined.

```
ExtractedFacesOfTheBody=ExtractedShellOfTheBody− >faceList();
ExtractedFacesOfTheBody ! = NULL AND
ExtractedFacesOfTheBody->geometry()− >identity()! =SPLINE_TYPE
```

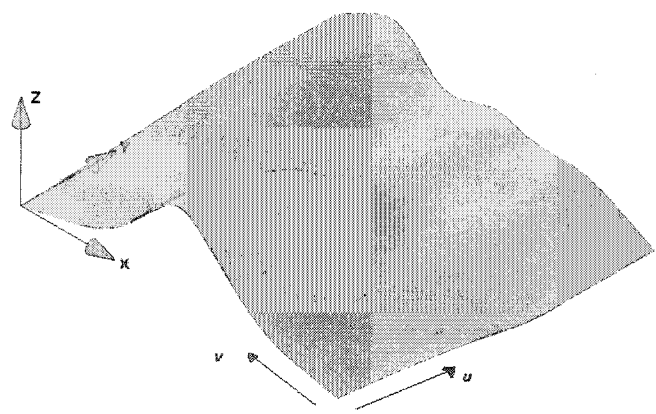
To ensure the versatility of the implementation, the system then interrogates the surface to ensure that its bounded. Since the different CAD systems may choose different parametrization methods (Piegl and Tiller (1997)), checking the parametrization range must be carried out to ensure that the evaluation algorithms will deal with the relevant areas of the workpiece.

In figure 3.2(a), the surface parametrization concides with the x-direction with the parameter u, while in figure 3.2(b), the surface parametrization is reversed.

The developed algorithm deals with the parametric representation of the surface,



(a) u-direction in x direction



(b) u-direction in -y direction

Figure 3.2: Surface mapping in parametric and Cartesian spaces

but the machine tool deals only with the Cartesian coordinate system. Thus, in order to ensure the right movement of the cutter evaluated in the path planning phase of this work, the system must check the workpiece orientation.

3.3 Curvature Evaluation

After making sure that the given surface model is consistent, the interrogation of the surface starts with the evaluation of the curvature values at each evaluation point. The evaluation positions as defined in equation (2.15), should be regular points ($S_u \times S_v \neq 0$). Where S_u and S_v are the partial derivatives of the surface S in the directions u and v. In the context of derivatives of the surface, the interrogation methods commonly used to evaluate the surface are classified according to the order of the derivatives used in the evaluation. Thus a method is called of n^{th} order if the evaluation uses surface/curve derivatives of this order. Second order surface interrogation methods will be used in this evaluation, since the surface must be C^2 continuous.

In this work several interrogation methods were adapted. All these methods are related to the curvature, which is the second order derivative of the surface. In the literature, different curvature measures were used for surface evaluation (Barnhill et al. (1988), and Farin (1993)). The principal curvatures, which are the extremes of the curvature values at this evaluation position k_{max} and k_{min},were first used in the evaluation of the surface properties. Since the interest was in constructing tool path using iso-parametric curves as a starting point, this method was not suitable for this application since the values of the principal curvatures may occur in arbitrary directions (Figure 3.3).

The Gaussian curvature, as defined in equation 3.1, which is the product of the principal curvatures, gives a better idea about the nature of the surface at the evaluation position.

$$K_{gauss} = k_{max}.k_{min} \tag{3.1}$$

The Gaussian curvature is a scaler value which makes it independent of the iso-parametric directions. The drawback of this measure is in tool selection phase of this

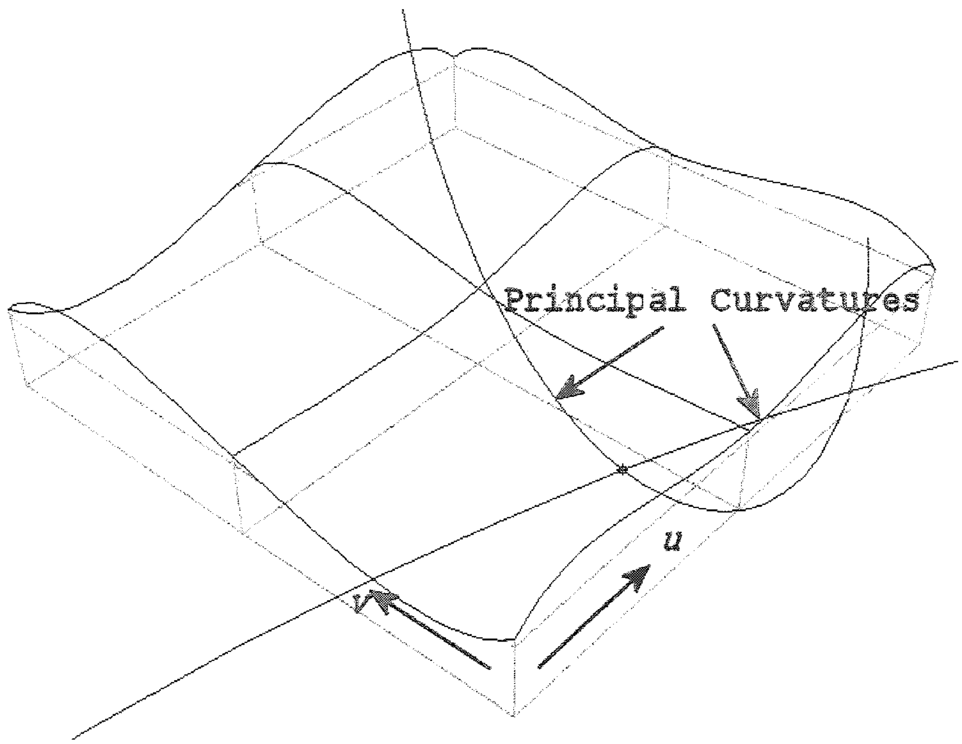


Figure 3.3: Principal curvature

work the direction of the curvature is of importance to the algorithm. Second, from the definition of the Gaussian curvature (3.1), certain classes of free–form surfaces (e.g. ruled surfaces) gives uniform value of zero all over the surface.

The following curvature property which was considered for this work is the mean curvature K_{mean}, as defined in equation 3.2.

$$K_{mean} = \frac{k_{max} + k_{min}}{2} \tag{3.2}$$

Since the mean curvature is the average of the principal curvatures, its value will lack the directional indications required for the different phases of this work.

The normal curvature is then considered for this surface interrogation. The value normal curvature (k_n) is derived from the normal to the surface at the evaluation point as shown in figure 3.4. The components of the normal to the surface is the result of the normal to the iso-parametric curves passing through the evaluation point (Equation 3.5). Then by evaluating the normal curvature of the iso-parametric curve in the u and v direction, the desired surface properties can be investigated.

$$N = \frac{S_u \times S_v}{|S_u \times S_v|} \tag{3.3}$$

$$k = \frac{d\mathrm{t}}{ds} \tag{3.4}$$

$$k_n = \frac{k}{N} \tag{3.5}$$

The normal curvature was used to extract several pieces of information. First by finding the curvature value in v direction, the critical sections in the surface are

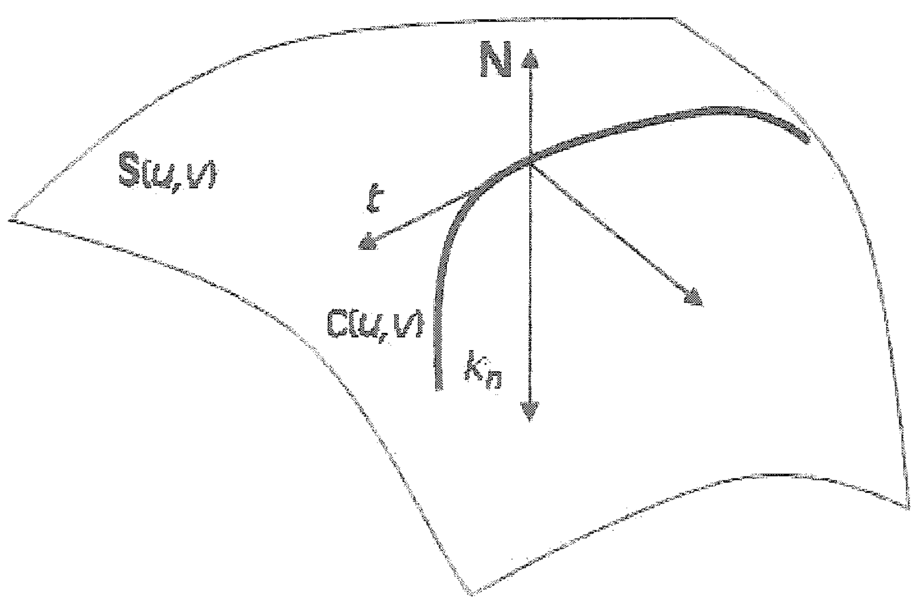


Figure 3.4: Normal curvature of 3D curve on surface

highlighted and ranked. The maximum curvature position is then found.

Since the aim of this curvature analysis is the aid in cutting tool positioning and tool selection, only the concave curvatures in this parametric direction are of interest($k_n^v < 0$).

The evaluation of the normal curvature in each direction is carried out by creating a "slicing" plane. This plane is intersected with the workpiece surface creating iso-parametric curve. Along this curve the parametric evaluation points are mapped to the Cartesian domain, then the corresponding curvature values are investigated.

The value of the maximum curvature in v direction is then used to constrain the tool diameter $D_{tool} = \frac{2}{k_n^v}$.

The next step is to evaluate the curvature in the other parametric direction. The algorithm to carry out this evaluation is the same as mentioned in the other parametric direction.

The importance of this evaluation is that it will determine the maximum permissible value for the fillet or insert radius.

3.4 Curvature Change Evaluation

The normal curvature values investigated in the previous section are very important in pointing to the problem areas on the workpiece surface (e.g. Jensen et al. (1995), and Jensen et al. (2002a)). Another measure that would address some surface properties of importance to the machining process that the curvature may miss is the curvature change ($\triangle k_n$). The curvature change at point p is defined in equation 3.6.

$$\triangle k_n(p) = |k_n(p) - k_n(p+1)| \tag{3.6}$$

The curvature change is a more comprehensive parameter in relating to the cutting tool posture change (tool slope). This is very critical to the tool path planning process since a sudden tool movement is not desired. Another reason for the curvature

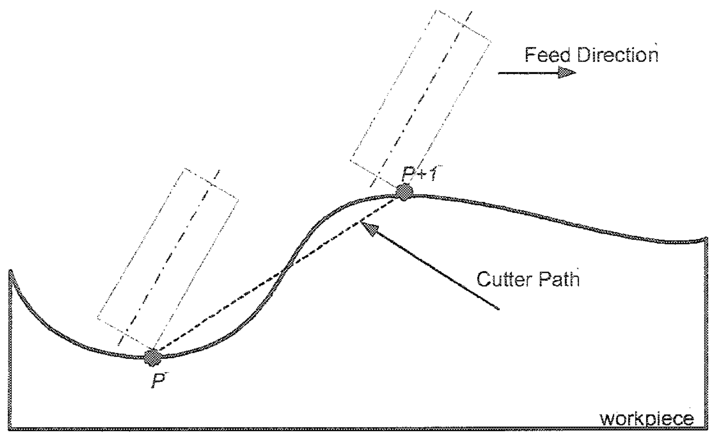


Figure 3.5: Curvature change in tool path planning

change to be of interest is the fact that the tool movement used to create the tool path for free–form surfaces are always approximated into straight line segments. To keep the surface smooth and within the given machining tolerance, special attention should be given to the parts of the surfaces that has the greater curvature change values.

Figure 3.5, demonstrates the importance of curvature change as surface interrogation. Point p has high curvature, while point $p+1$ has lower curvature which puts it in a lower rank in the evaluation algorithm. From the point of view of curvature–based methods the forward step between the two points does not need more refinement. Meanwhile the curvature change algorithm will detect the change of the normal direction between the two points and its value will rank high in the algorithm priority. This will trigger further investigation in this span and the chord length deviation algorithm will be applied to reduce the forward step.

The curvature change investigation is carried out by inspecting all the evaluation points. The evaluation is carried out in both forward and backward directions. The curvature change points are then sorted and the maximum curvature change is then used as an initial point for the tool path error checking in the next step.

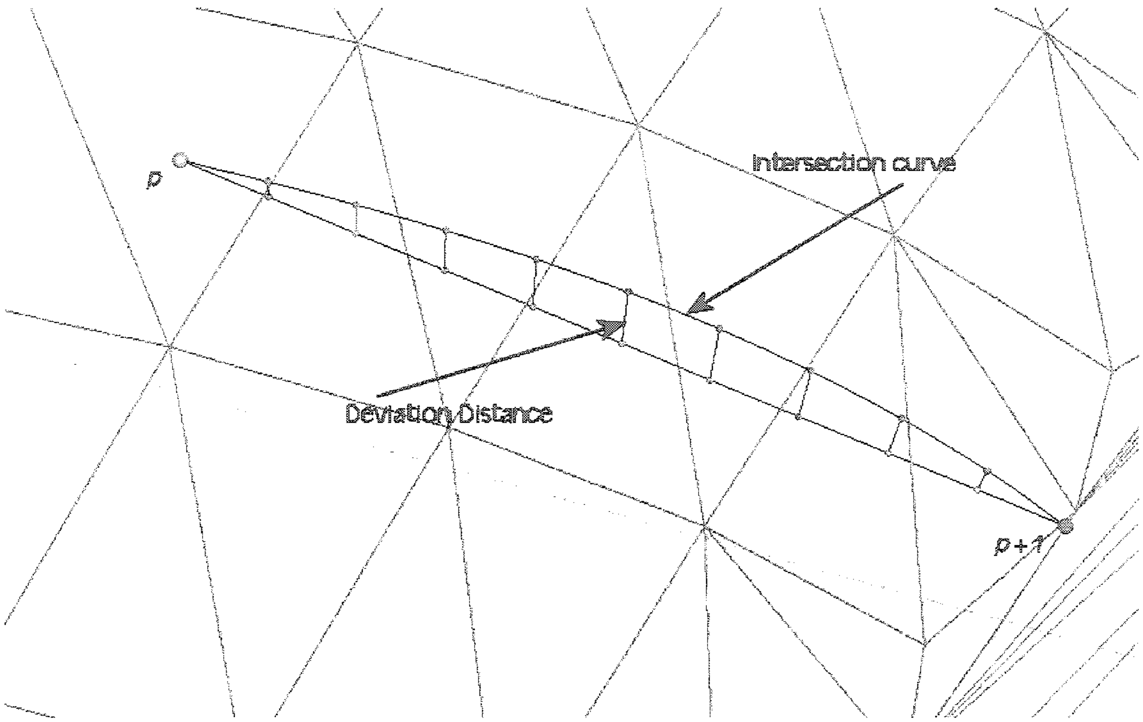


Figure 3.6: Chord length deviation

3.5 Chord Length Deviation Checking

In order to optimize the number and locations of the evaluation points on the surface, a chord length deviation check algorithm is performed. The span between the maximum curvature change point and the adjacent points is subdivided into sub evaluation points (see figure 3.6). The distance between the deviation line and the surface is then inspected. To make sure that the deviation distance is always measured in the desired direction, the evaluation is carried out between the deviation line and an iso-parametric curve created by the slicing plane mentioned earlier.

The result from this stage is compared to the value of the given machining tolerance. The forward step is then determined according to this result. If the deviation is greater than the permissible tolerance value then the forward step is reduced and the evaluation process is restarted from the beginning.

3.6 Summary

In order to optimize and simulate the cutting process, The workpiece surface is investigated. Starting from the iso-parametric tool path evaluation method, curvature related smoothing and error compensation algorithms were developed. The normal curvature was used to collect the data needed for the tool positioning and tool selection phases of this work. The curvature change algorithm was used to pinpoint the critical parts in the tool path and for guiding the refinement and smoothing algorithms. The output from these algorithms are needed for the tool modeling and the tool positioning phases that will be carried out next.

CHAPTER 4
GENERIC TOOL MODEL

4.1 Introduction

The development of the simulation system for the milling process required the definition of a mathematical and/or a geometrical model for the workpiece. In order to complete the modeling of the system, the cutter modeling problem needs to be addressed.

To create a robust simulation process that deals with different configurations of the process in a unified way, a generic tool model was developed. In this field different mathematical and geometric models were developed to define the key features in the most frequently used cutting tools. The degree of the sophistication of the cutting tool model is dictated by the amount of data needed to perform the task at hand. In some cases the simulation of the cutting edge is the most important, while in other cases the simulation of the tools outer profile is required to complete the simulation. Hybrid models are sometimes used to integrate different cutting process simulation and optimization modules.

In the current work, the development of new generic tool model was carried out. Developing this work using the geometric kernel allowed for certain improvements over the models found in the literature (e.g. Lee (1998b), Karunakaran et al. (2000)). Some models require ten or more parameters to define the cutting tools, while other models requires only eight parameters. The number of parameters is a function of the purpose of the simulation. In the following sections an overview of the developed system will be presented and some illustration of this phase's output will be shown.

4.2 Geometric Model Creation Process

A geometric modeling approach was chosen to represent the cutting tool in this work. The sufficient details needed for tool path simulation were investigated. The model represents the outer envelope of the cutting tool. This allows the model to be used for tool position checking, gouge detection and collision detection. The solid model is also used for updating the workpiece stock to check the surface errors.

The tool model is created by revolving the tool profile about the tool axis. In order to discuss the model creation, the variables utilized in the process are shown in figures 4.1 and 4.2.

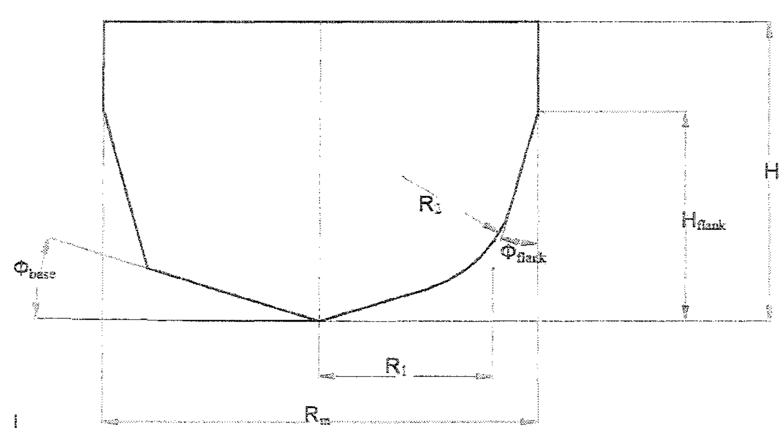


Figure 4.1: Generic tool model parameters

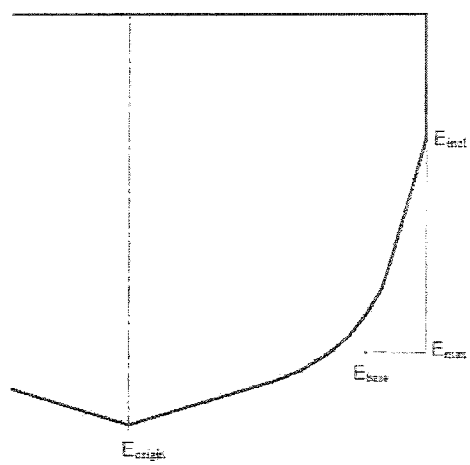


Figure 4.2: Tool model Construction points

4.3 System Overview

The developed model is to be used in optimization and verifications processes performed in different phases of this work. Thus the logical beginning point for this phase was to seek the constraints imposed on the model up to this point.

The data resulting from the workpiece evaluation phase gives the maximum allowed tool diameter, the maximum allowed fillet radius and, the minimum tool length. These values will change and the tool model will be updated in later stages of this work.

As shown in the flow chart detailed in figure 4.3, after retrieving the model constraints, user input is required to define the actual model parameters. This interaction is required in order to improve the system by using standardized tool data. The system uses the retrieved constraints to bound the user selection and make sure that all requirements are met before proceeding with the geometric creation models. The input data given by the user is then used to create the tool profile. This process will be discussed further in the following sections.

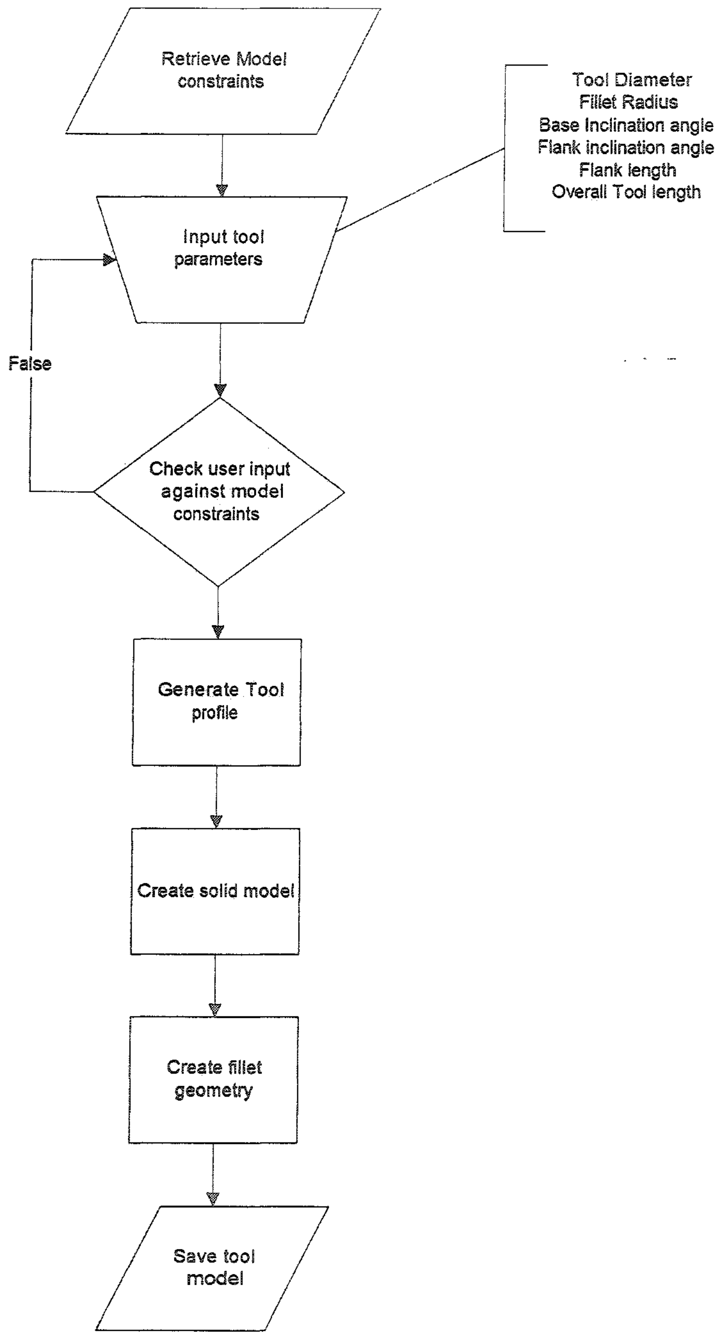


Figure 4.3: Generic tool model

4.4 Tool Profile Definition

The model developed in this work needs six independent parameters to create the tool profile. These parameters shown in figure 4.1 are, tool diameter (R_m), fillet/insert radius (R_2), tool base inclination angle (Φ_{base}), tool flank inclination angle (Φ_{flank}), length of inclined flank (H_flank) and, overall tool length (H).

From these parameters the points needed to generate the tool profile are created as shown in figure 4.2. The tool origin point E_{origin} is taken as the coordinate system origin (0,0,0) to facilitate further transformations during the next phases of the work. The point $Emax$ is then defined in equation 4.1.

$$E_{\max} = \begin{pmatrix} E_{origin}.X + R_m \\ E_{origin}.Y \\ E_{origin}.Z + R_m.\tan\phi_{base} \end{pmatrix} \tag{4.1}$$

The intersection point E_{base} is then created to define the point needed for base and flank lines creation as in equation 4.2.

$$E_{base} = \begin{pmatrix} E_{\max}.X - H_{flank}.\tan\phi_{flank} \\ E_{origin}.Y \\ (E_{\max}.X - (H_{flank}.\tan\phi_{flank}))\tan\phi_{base} \end{pmatrix} \tag{4.2}$$

In order to define the end of an inclined part and the beginning of the straight part, the point E_{incl} is then defined in equation 4.3.

$$E_{incl} = \begin{pmatrix} E_{origin}.X + R_m \\ E_{origin}.Y \\ E_{\max}.Z + H_{flank} \end{pmatrix} \tag{4.3}$$

After the definition of these points the tool profile creation process can be started.

4.5 Tool Profile Creation

The geometric constraining of the profile was carried out by defining the tool parameter and key points. The data is then used to create wire bodies. First using E_{origin} and E_{base} the straight part of the tool base is created. The inclined part of the tool flank is then created using E_{base} and E_{incl}. The straight part of the tool's shank is then created using the tool overall length data.

A wire body is generated by joining these lines. This wire body is swept around the tool axis to create a solid body.

As shown in the left side of figure 4.1, the generated tool would take this shape. To create the tool fillet/insert geometry a healing process should be performed next. To do that the intersection curve between the tool base and flank should be extracted. This curve is found using a search algorithm in both directions to find the co-edges on both surfaces that belongs to the intersection curve. After finding the common edge, the fillet data is used to create the fillet geometry using healing process. By using the healing process, the model consistency is ensured and the transition between the surfaces constituting the the tool model is guaranteed to be smooth.

4.6 Generic Tool Geometries

The developed tool model can define several milling cutter geometries. This was meant to increase the flexibility of the tool selection process that will be performed in the next phase. The model can define flat, filleted, ball, conical end mill cutter, in addition to defining twist drills and elephant-foot end mill geometries. Figure 4.4, illustrates the different capabilities of the modeling system developed.

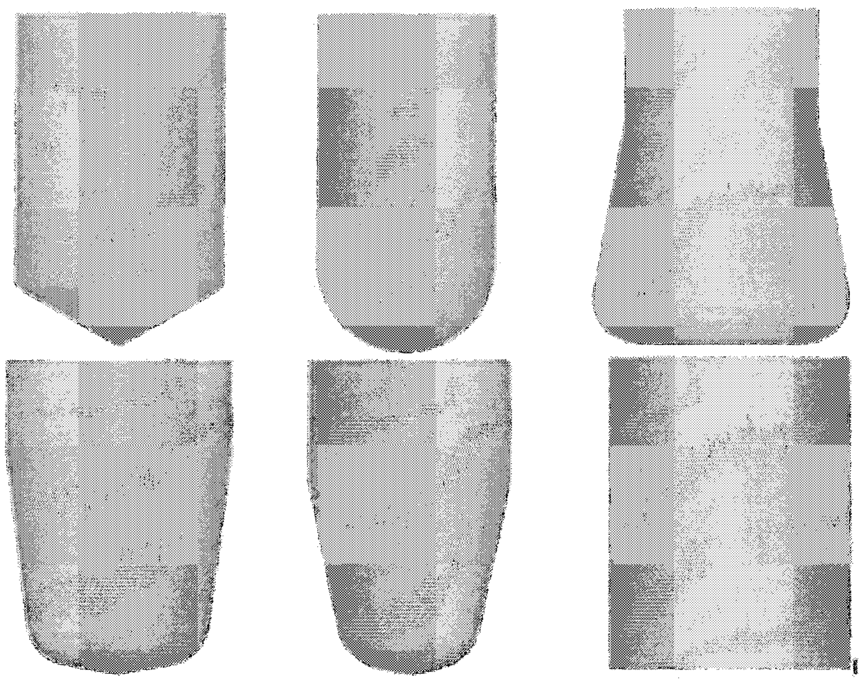


Figure 4.4: Tool modeler output examples

4.7 Summary

In order to perform the different simulation and optimization tasks, a generic tool model was developed. This model can define several cutter types. The developed model is defined by six independent parameters. This gives it an advantage over some geometric tool models found in the literature by using less parameters. The model is based on a solid model and is refined based on the output information from the different optimization modules. A detailed description of this module's algorithms be is found in appendix A.2.

CHAPTER 5
TOOL ORIENTATIONS AND STEP-OVER DISTANCE EVALUATION

5.1 Introduction

In multi-axis milling the tool orientation always varies to ensure certain contact between the tool and the part. This is essential to avoid the creation of surface errors. Each tool posture is described by its tool tip coordinates (x, y, z) and tool orientation (a, b, c), expressed in the part coordinate system in most cases.

The tool positioning problem can be classified based on the cutting tool type into two categories. The positioning of the ball end mill, and the positioning of other cutter types. The reason for this classification is that most of the positioning methods use the effective tool profile, or tool silhouette, to evaluate the tool orientation. The ball end mill effective profile is a circle regardless of its orientation. Therefore, for the ball end mill, the positioning problem is simplified into accessibility and collision avoidance problem. On the other hand the tool positioning problem for other types of end mill cutters is dependent on the effective cutting profile and related to the workpiece topology. The optimal tool positioning strategy is to select the maximum tool size that will match the surface topology and result in minimum deviation from the design geometry.

In this work workpiece topology and tool geometry are used to optimize the tool orientations to eliminate surface error. Tool step-over is then evaluated by controlling the cusp height between adjacent tool paths. The general layout of this algorithm is shown in figure 5.1. Detailed flowcharts can be found in appendix A.3.

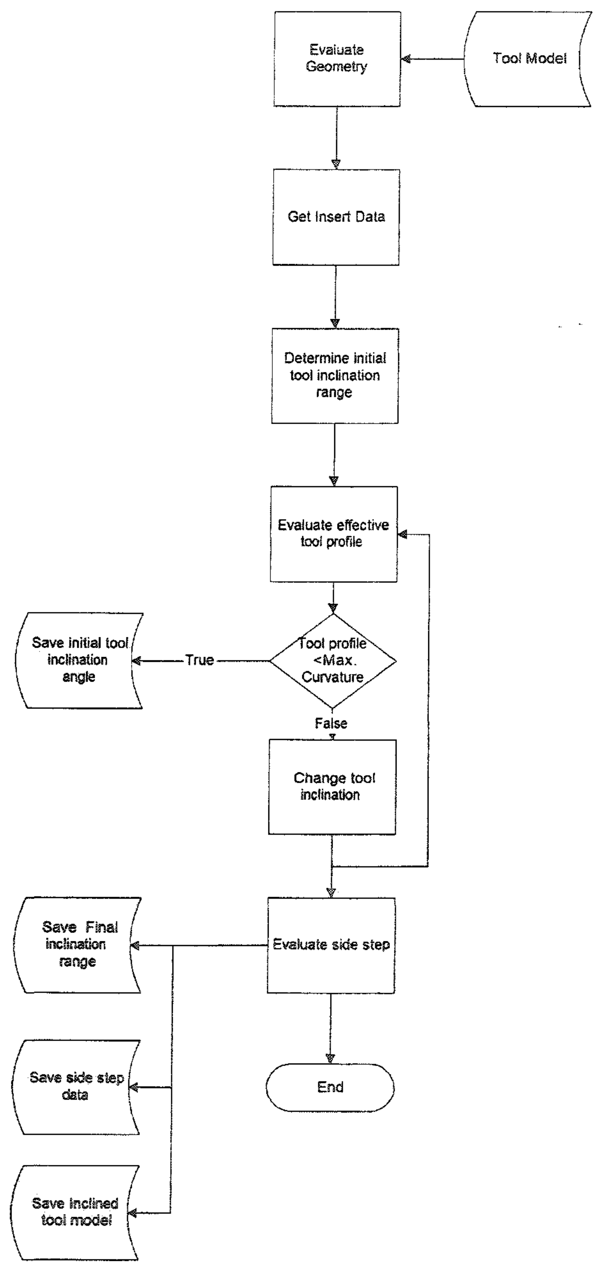


Figure 5.1: General layout for tool orientation and step-over evaluation algorithm

5.2 Tool Orientation

In five axis machining using filleted or flat end mill to machine free-form surfaces, it is a common practice to keep a fixed inclination angle of the tool relative to the workpiece surface (Jensen et al. (2002a)). This pre-inclination angle is referred to as the Sturz angle and the milling operation performed using a fixed inclination angle is always referred to as Sturz milling. The Struz angle is typically between 5° to 10°. The angle is chosen so as to minimize the probability of back gouging and cutter's bottom contact with the machined surface. The use of fixed inclination angle limits the use of the filleted or flat end mill cutter to a single effective tool profile, thus limiting the tool choice to single tool diameter. In this work the tool inclination is allowed to vary according to the local surface topology, which increases the tool utilization and reduces the surface errors. Figure 5.2 shows the different terms that will be used to describe the tool's inclination during this work.

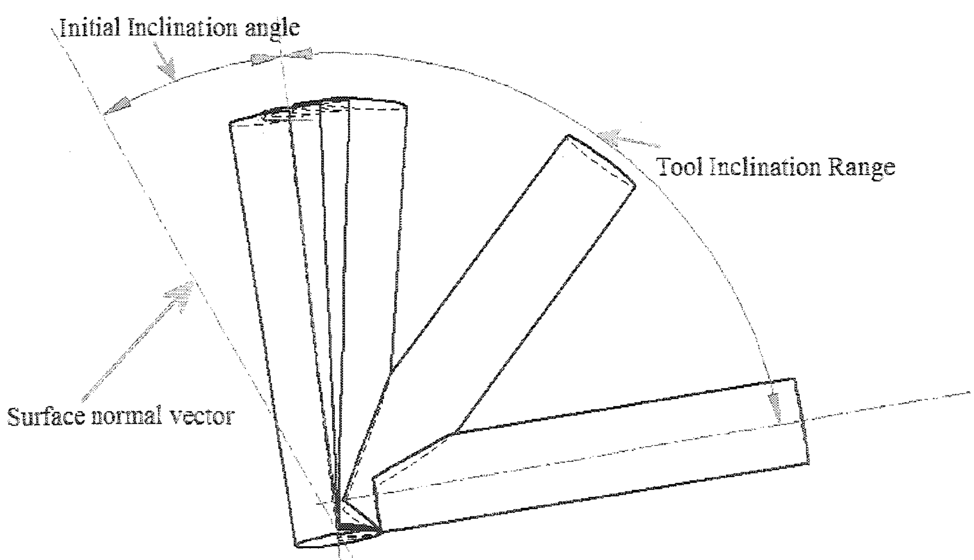


Figure 5.2: Tool inclination range

5.2.1 Initial Tool Inclination Range

The initial tool model created in the previous phase of this work possesses geometrical properties, which are of importance to the tool path optimization process. Due to the ball end milling process inefficiencies (Vikers and Quan (1989), Jensen and Anderson (1992)), different end mill geometries were used. To maximize the tool utilization, the tool contact with the workpiece should be kept as large as possible. This process is bounded by the workpiece topology. In order to avoid positioning errors, two considerations must be kept. The first consideration is to keep the tool contact point within the cutting (insert/ fillet) part of the cutter. The second consideration is that the tool profile should fit inside the workpiece surface at all times.

The main advantage of using various end-mill types is to take advantage of its variable effective tool profile properties. This can be achieved by altering the tool inclination and tilt angles. The inclination angle is defined as the angle between the tool axis and z-axis in the local coordinate system projected in the feed direction plane. While the tilt angle is defined as the angle between the tool axis and z-axis as well, but projected in the plane normal to the feed direction.

The inclination of the cutting tool reflects on its effective profile. The tool profile is almost a straight line in the upright position for the flat or filleted end mill, while the profile tends to match the ball end mill shape in its ninety degrees position.

It can be seen from the tool geometry that the inclination range depends on the start and end angles of the cutting part of the tool as shown in figure 5.3. This range is called the initial inclination range. The relation between the tool inclination angle and fillet/insert angle is given by equation 5.1.

$$\lambda_{incl} = (\pi + \psi_i)\,, \psi_s \leq \psi_i \leq \psi_e \tag{5.1}$$

The algorithms developed in this module begins by retrieving the created tool model. In the model creation process several smoothing and blending algorithms were

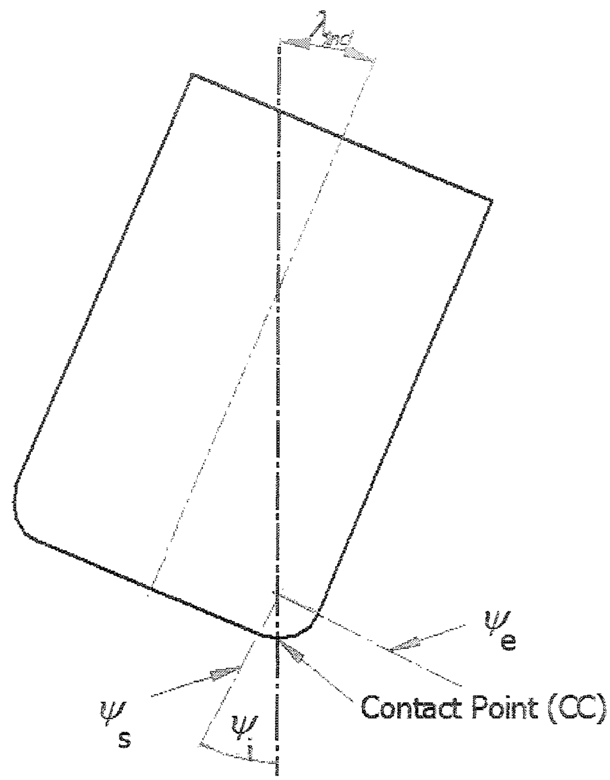
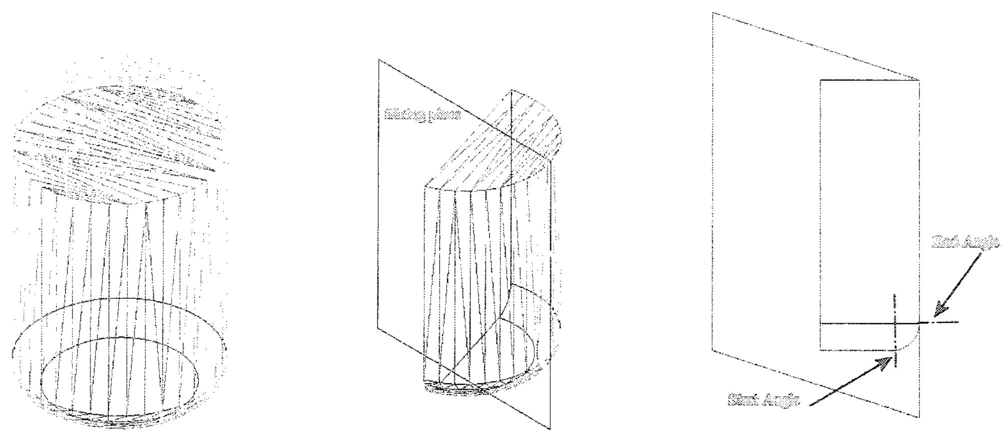


Figure 5.3: Tool inclination angle relation to fillet/insert angles

used. This allowed for the reduction of the number of parameters required to generate the model. On the other hand, the fillet data required for the investigation of the tool inclination is not readily available and must be extracted from the model. To carry out this task a slicing plane, that splits the tool model into two halves, was created. The reasons for investigating half the tool model is that, firstly, only the leading part is required for the tool orientation investigation. Secondly, using the search algorithm to find the tool insert data will yield two results for the leading and trailing parts of the tool, due to the symmetrical properties of the model. To eliminate this, splitting operation was necessary (see figure 5.4).

After splitting the tool, the profile section is needed to get the fillet start angle, end angle, and center.

The initial inclination range is now determined and can be used as a guideline for effective tool profile evaluation.



(a) Generic Tool Model (b) Sliced Tool (c) Fillet/Insert Data

Figure 5.4: Tool data extraction

5.2.2 Effective Tool Profile Evaluation

In five axis machining, the orientation of the cutting tool varies to match the surface topology. Based on the tool/workpiece constraints, a tool positioning strategy with variable tilt and inclination angles was developed to keep the cutting tool contact with the workpiece surface in a position that minimizes undercuts.

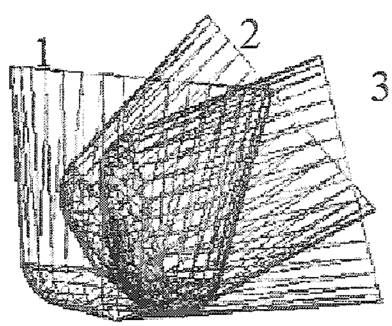
The tool/workpiece profile matching ensures that the maximum contact length is utilized, while reducing machining errors.

The effective tool profile is defined as the effect that the tool imprints on the workpiece while passing through the contact point. Several methods were used to obtain the tool profile. These methods are mostly mathematical (e.g. Cho et al. (1993), and Lee (1998b)). The advantage of using the mathematical approach is that less computional time is required to complete the task. On the other hand in order to derive the mathematical expressions needed for evaluations, some approximations must be introduced to reach the final form. The geometrical evaluation used in this work utilize the solid model of the tool and workpiece. The computional effort to extract the effective tool profile is more intensive than the mathematical approach, but the resulting space-curve representing the effective tool profile is a parametric curve that can be dealt with in model space.

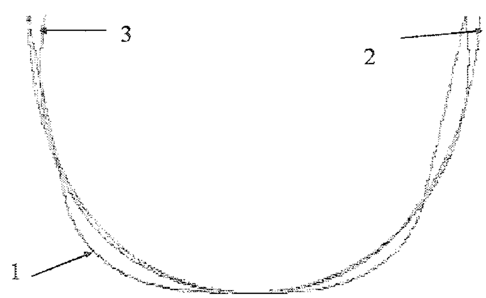
In order to carry out effective tool profile evaluation, an intersection plane was created passing through the tool contact point and in the normal direction to the feed direction (equation 5.2). The plane is then evaluated against the tool model and the resulting intersection curve is then saved for feasible inclination range evaluations.

$$T_p = \left\{ \begin{array}{c} M_{cc} \\ N \end{array} \right\} \tag{5.2}$$

The effective tool profiles resulting from various tool inclinations are shown in figure 5.5.



(a)



(b)

Figure 5.5: Effective tool profile

5.2.3 Feasible Inclination Range Evaluation

The initial inclination range as described by the tool geometry cannot *yet* be used in tool positioning. This was due to the variable surface topology of the workpiece. In order to take full advantage of the tool, the effective tool profile(s) must be evaluated against the workpiece topology. The tool profile must fit in or *match* the workpiece profile at the evaluation position. This ensures the elimination of surface errors due to tool positioning.

In order to do so, the instantaneous workpiece profile at each evaluation position is created by using the intersection plane defined in equation 5.2. The intersection evaluation is carried out against the workpiece model this time.

To obtain the feasible inclination range, Boolean evaluations of the tool effective profile is performed at each position against the instantaneous workpiece profile. This operation then goes through the whole initial inclination range eliminating the interfering tool profile from the array of inclinations. Figure 5.6 shows the different regions of rejected and feasible inclinations obtained for a case study.

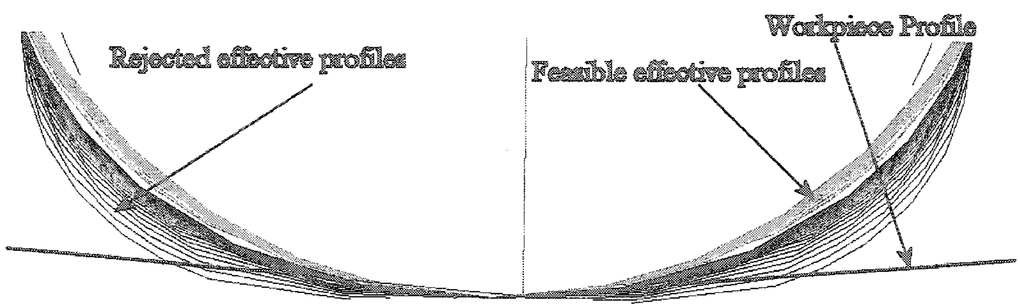


Figure 5.6: Feasible inclination range

During the evaluation of the feasible inclination range, the predefined value of the pre-inclination angle is checked. If the first feasible inclination angle is larger than the pre-inclination angle, then the tool model is considered rejected by the algorithm. In this case the algorithm will stop the system from proceeding to the next step unless the tool model is modified and the tool diameter is reduced. After the creation of the tool model the effective tool profile and feasible inclination range procedures must be carried out again. The feasible inclination range is then saved for the tool path generation and verification phase that will be carried out in the next module.

5.3 Step-over Distance Evaluation

Free Form surface machining using three axis ball end mill was widely used (e.g. Choi and Jun (1989), Vikers and Quan (1989), Zhou and Lin (2001)). Using a ball end mill produces cusps between the adjacent tool paths. To reduce the cusp height, the step—over distance must be reduced, which increases the tool path length. As mentioned in the previous section the use of different end mill geometries maximizes the utilization of the tool profile and reduces the need for a more intensive tool path.

The problem of determining the step-over distance in three axis ball end milling was discussed in section 2.4.3. The constant circular profile of the ball end mill makes it possible to derive the mathematical expression for the relation between the step-over distance and cusp height for certain tool diameters. On the other hand, using flat or filleted end mill results in variable effective tool profile. This makes the determination of step-over distance a more challenging task.

The most accurate method to calculate the step-over distance as a function of the cusp height is to evaluate the tool profile at each adjacent tool position. This method, while accurate, requires a huge amount of calculations given the fact that the variable tool profile prevents the derivation of a mathematical formula to represent the whole tool path, thus a geometrical evaluation was used to resolve this situation.

By evaluating the step-over distance at each tool path point, the resulting step-over distance will vary not only between adjacent tool paths but from position to position along the same tool path segment.

In order to speed up the evaluation while maintaining the cusp height within the given tolerance, the step over evaluations are performed at the critical parts of the workpiece, known to have the maximum curvatures that can result in higher cusp height.

The developed algorithm retrieves the workpiece data from the curvature evaluation phase. Feasible inclination data is then used to orient the tool in two adjacent tool path positions. The initial step-over distance was taken as the tool radius to ensure the maximum overlap between the two adjacent positions.

The solid tool models in the two adjacent positions are then joined and a solid instantiation marching algorithm advances the tool (Figure 5.7). Boolean subtraction of the tool from the workpiece stock is carried out at each step.

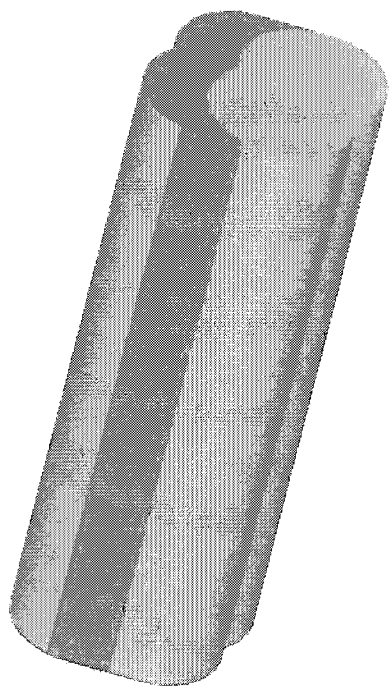


Figure 5.7: Tool model instantiation and marching

The resulting workpiece segment at the evaluation position is then analyzed by the creation of an oriented bounding box. The height of the bounding box indicates the cusp height resulting from the current step-over distance. Figure 5.8, shows the resulting workpiece segment and the cusp height associated with it.

The algorithm then compares the resulting cusp value with the given cusp height and tries to change the step-over distance to converge toward the given cusp height. As the algorithm tries to reach a value closest to the given constant, it is impossible,

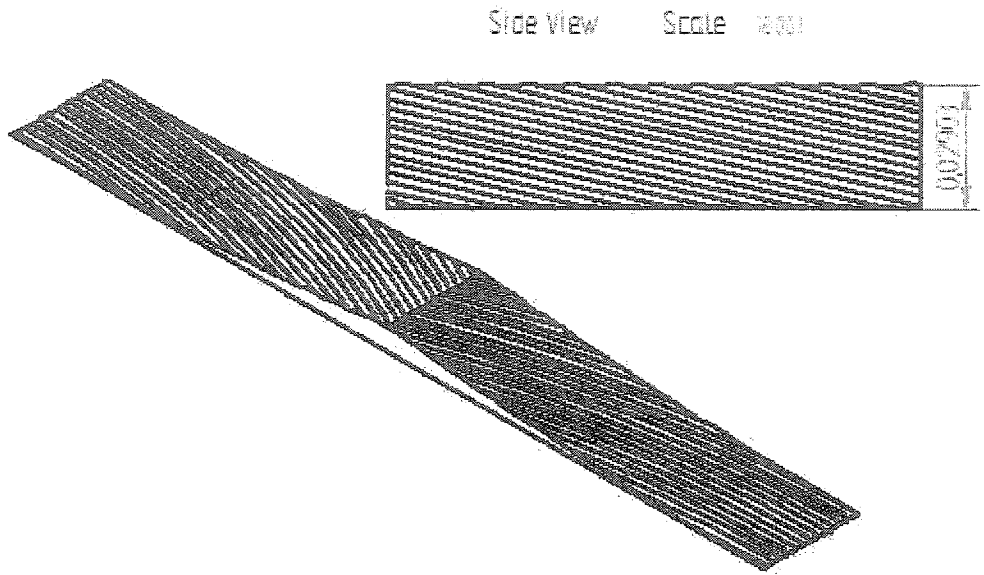


Figure 5.8: Cusp height evaluation

due to the variable tool profile properties, to obtain the exact value. A certain tolerance zone is taken as a percentage of the given cusp height value to allow for the algorithm to terminate, otherwise the computional time may extend indefinitely.

By obtaining the step-over distance the data needed to generate the tool path is almost complete. The step-over distance is then saved to be used in the following phases of this work.

5.4 Summary

In 5-axis machining the tool path consists of both the tool coordinate (x, y, z) and tool orientation (i, j, k). To create a usable tool path the workpiece topology should be taken into consideration. Depending on the tool geometry a feasible inclination range was created to guide the tool positioning. A checking algorithm was used to ensure that the tool posture matches the required shape of the workpiece to avoid undercuts. The developed algorithms ensures that the system will not proceed to the tool path generation and checking phase if the tool is not acceptable. If a given tool did not pass this phase, the tool model refinement is required to create a tool with smaller diameter and the re-evaluation of the effective tool profile must be carried out again.

The step-over evaluation ensures that the required cusp height is obtained by the recreation of the cutting process at the critical tool path positions marked by the maximum curvature points.

The resulting data from this phase is used in the following tool path optimization phase.

CHAPTER 6
TOOL PATH GENERATION AND GOUGE AND COLLISION EVALUATIONS

6.1 Introduction

The generation of a tool path for five axis machining with flat or filleted end mills requires paying attention to the optimization of several parameters. These parameters include the tool shape, orientation and movement. In addition to these parameters, the tool path should take into account the errors caused by the tool size and movements.

In the previous phases of this work, tool shape was selected and refined to comply with the workpiece topology constraints as with variable tool profile characteristics. The tool path forward steps were previously optimized to comply with machining tolerance and to smooth tool transitions. The step-over distance was optimized to generate the required cusp height.

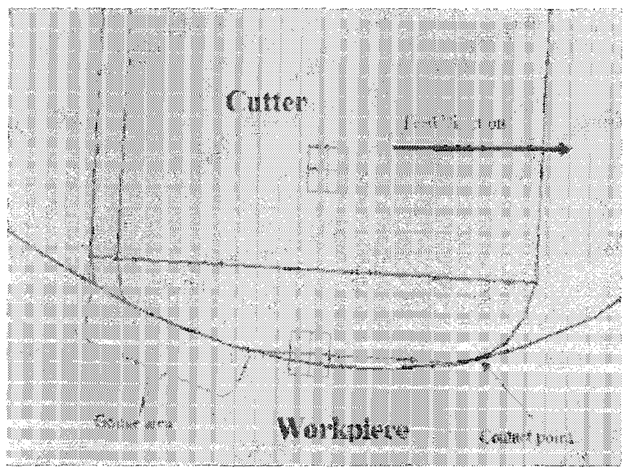
The tool path can be generated using workpiece topology data and the tool path optimization output so far. The tool position and orientation based on this data should eliminate surface errors due to tool positioning.

Although the generated tool path should produce the desired surface topology, other considerations arise that may cause the generated surface to be out of specifications. Two factors must be taken into account in the optimization of machining path, these factors are tool rear gouge and collision. The tool may touch the workpiece in a position other than the intended cutter contact point (Figure 6.1(a)), which causes the finished surface to be re-machined. This area is considered a gouge area and must be dealt with to generate an optimized tool path

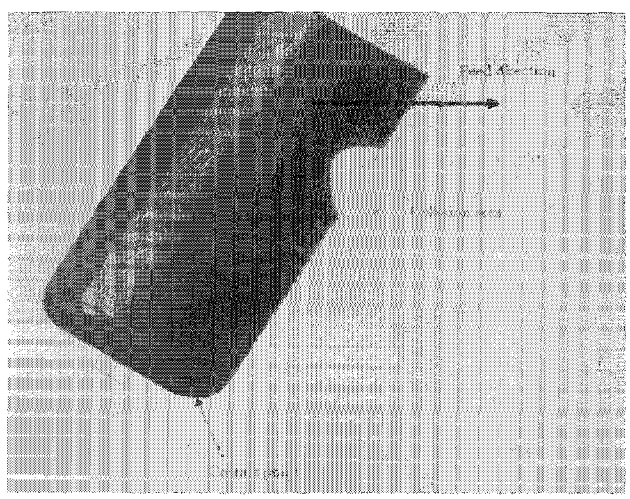
The second situation is that the tool shank or holder may get in contact with the workpiece (Figure 6.1(b)). This may cause tool breakage, machine damage or the prevention of the tool from reaching certain planned tool path areas. This situation must be resolved to prevent machine tool damage and to generate an optimized tool path.

Figure 6.2, illustrates the general layout of the developed algorithm. More detailed flowcharts can be found in appendix A.4.

In the following sections the tool positioning process as well as the rear gouge/collision evaluation process will be discussed. Error detection and compensation strategies will be reviewed, and a tool modification procedure will be explained. By the end of this chapter, the tool path generation methodology will be completed.



(a) Rear tool gouge



(b) Projection of C space

Figure 6.1: Tool gouge and collision definitions

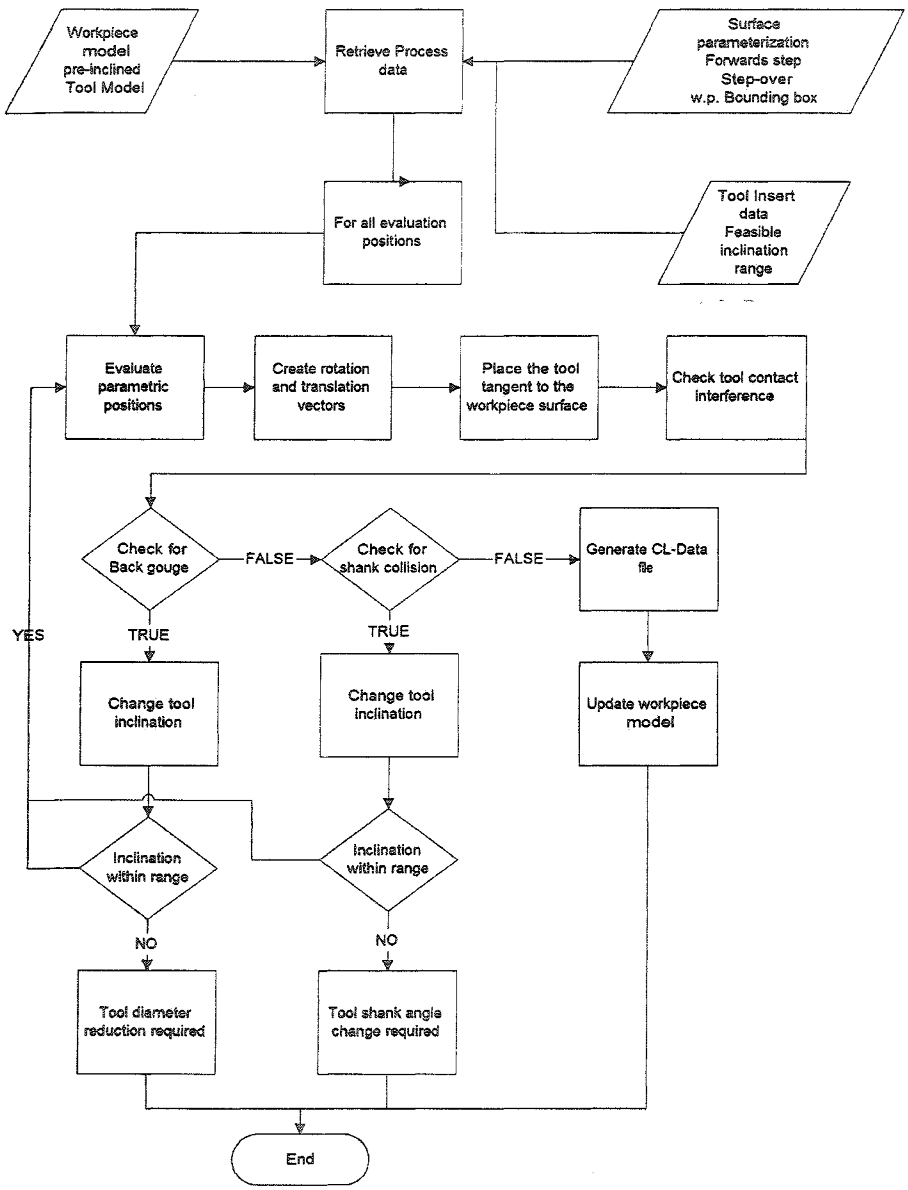


Figure 6.2: Tool positioning, gouge and collision evaluation.

6.2 Tool Positioning

In this tool path generation phase, the tool must be placed correctly relative to the workpiece surface. In order to be sure that the tool position and orientation is correct, the tool movements should be controlled to prevent the formation of undercuts.

The tool orientatins must be given special attention in this phase. Since various tool types are used in five axis machining, the tool/workpiece contact point should be extracted from the tool model. The tool's rotation point varies with fillet/insert geometry. The developed algorithm takes into consideration the output data from the previous phases to evaluate the tool rotations.

The tool rotation is performed around a point created from the intersection of the normal vector to the workpiece surface (N) passing through the contact point (M_{cc}) as defined by the constraints in equation 6.1. The rotation axis should be evaluated according to the workpiece topology and tool geometry. This axis is created using the plane normal to the feed direction and the normal vector created to evaluate the rotation origin as defined by the constraints in equations 6.2 and 6.3.

$$M_rot = \left\{ \begin{array}{c} M_{cc} \\ N \end{array} \right\} \tag{6.1}$$

$$T_rot = \left\{ \begin{array}{c} M_{cc} \\ N \end{array} \right\} \tag{6.2}$$

$$\overline{S}_{rot} = \left\{ \begin{array}{c} M_{rot} \\ \perp T_{rot} \end{array} \right\} \tag{6.3}$$

This procedure insures that the tool insert/fillet surface remains in contact with the workpiece surface all the time without the creation of surface errors. Figure 6.3(a), shows the workpiece error due to tool rotation. If the tool rotation is performed

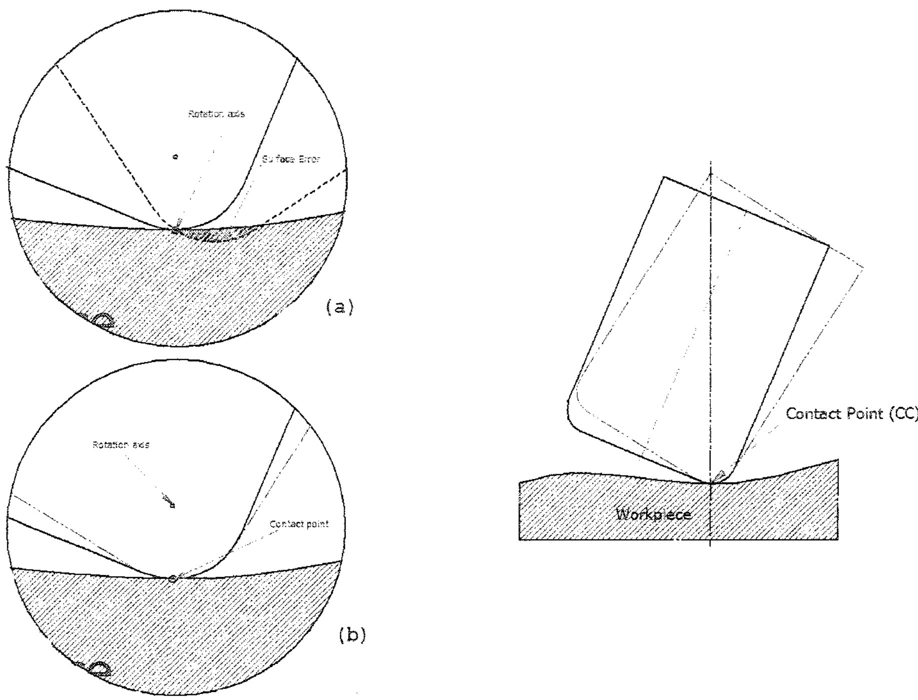


Figure 6.3: Workpiece error due to tool positioning

around the tool/workpiece contact point the tool may remove more material than the desired amount, thus creating surface error. The developed tool positioning algorithm utilized the fillet/insert and normal direction vector to ensure that this positioning error will be eliminated (Figure 6.3(b)).

6.3 Gouge and Collision Evaluations

The tool positioning criteria discussed in the previous section would not be complete without collision and gouge evaluations. Although the tool position with respect to the workpiece surface (contact point) was evaluated, the tool size and orientation may cause the tool trailing part to cut into the finished part and/or the tool shank to hit the part while trying to reach the desired area.

Several methods were developed to avoid collision and gouging. For example the calculation of the nearest distance between tool axis and the workpiece and comparing this value with the tool geometry to detect gouging and collision. Other techniques perform intersection checks mathematically and compensate for the error value. The accuracy and speed of these methods are greatly affected by the modeling and tool path simulation methods. Some techniques, for the purpose of increasing the calculation speed, approximate the workpiece by polyhedron or faceted surface. This simplifies the intersection check from a cylinder/ surface or solid intersections to a cylinder/ plane evaluation. The choice of the degree of simplification is characterized by the degree of accuracy required in the final product.

In this work, the tool model is evaluated against the workpiece model to check for gouge or collision by performing interference checks. To speed up the evaluation process, without sacrificing the accuracy of the evaluation, the tool model is divided in order to reduce the number of geometric entities involved in the evaluation. As shown in figure 6.4, Only the tool part relevant to the evaluation phase is involved in the checking. If the interference check indicates a problem, then further procedures to eliminate the gouge or collision are but into use.

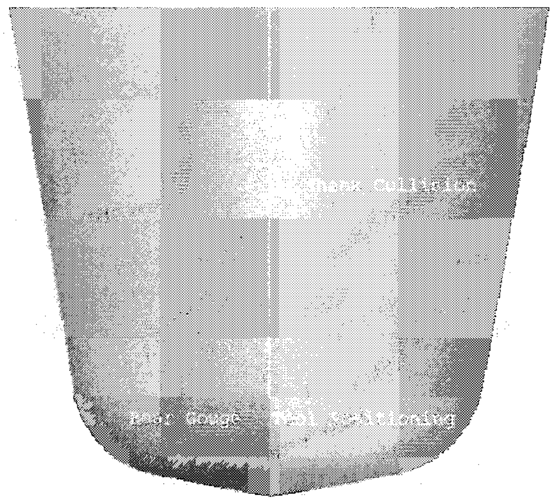


Figure 6.4: Tool model divisions

6.3.1 Back Gouge Detection and Avoidance Strategy

The rear gouge of the tool with the finished workpiece surface results in re-machining of the finished part of the workpiece. This tool path error is caused by the tool size. Large tool may cause this type of error due to the complex movement of the tool in five axis machining process.

To detect this problem, the developed work checks each tool path position while generating the tool path. The position in which gouge is detected must be evaluated thoroughly. The evaluation of the tool model together with the final workpiece model would give the indication and the value of the gouge. The tool inclinations are modified for this evaluation point to avoid the tool cutting into the finished surface. If the current tool parameters cannot satisfy these constraints further refinement in the tool diameter and the fillet/insert radius is required.

The part of the tool used in the gouge evaluation is the trailing part of the tool model. This is to reduce the interference calculations by limiting the number of elements involved to the relevant parts of the tool.

When interference checking indicates that the rear part of the tool is interfering with the finished part surface, maximum interference (deviation) algorithm is called. In the maximum deviation algorithm, a section is created by cutting the tool trailing part and the workpiece using parting plane. The parting plane is created using the tool axis and tool/workpiece contact point. The maximum tool/workpiece overlap is then calculated using vectors originating from workpiece intersection curve reaching to the tool intersection curve. The gouge-deviation vectors are created as normal vectors to the line connecting the deviation evaluation point on the workpiece intersection curve M_{dev_wp}) and the fillet-section center point M_{fillet} as defined in equation 6.4.

$$\overrightarrow{S}_{dev} \times \overrightarrow{S}(M_{fillet}, M_{dev\_wp}) = 0 \tag{6.4}$$

The maximum length of the gouge-deviation vector indicates the value of the gouge, while the direction is used to indicate the compensation direction (Figure 6.5).

The compensation algorithm is then called to eliminate the gouge. The tool inclination is altered by the angle ψ_{gouge}. The value of compensation angle is calculated from equation 6.5 as follows;

$$\psi_{gouge} = \tan^{-1}\left(\frac{\left|\overrightarrow{S}_{Dev}\right|}{|S(M_{fillet} - M_{Dev\_wp}|}\right) \tag{6.5}$$

The gouge compensation angle should be within the feasible inclination range $\psi_s \leq \psi_{gouge} \leq \psi_e$, otherwise the tool diameter should be reduced. If the gouge compensation angle was found acceptable, the tool positioning and checking algorithm must be re-evaluated for this position. While in the case that tool model modification is required, the operation should be restarted from the feasible inclination range evaluation procedure again, since the data obtained in this earlier phase is no longer valid.

If the tool diameter reduction is required, the value of the modified tool model should comply with the constraints in equation 6.6. The tool model will be modified as in figure 6.6, to eliminate the gouge situation.

$$\Psi_{tool} = \begin{cases} R_{1\_m} \leq R_1 - (L_{gouge}/2) \\ \psi_s \leq \psi_{Coll} \leq \psi_e \end{cases} \tag{6.6}$$

The frame work of this algorithm is presented in figure 6.7, for more detailed flowcharts see appendix A.4.

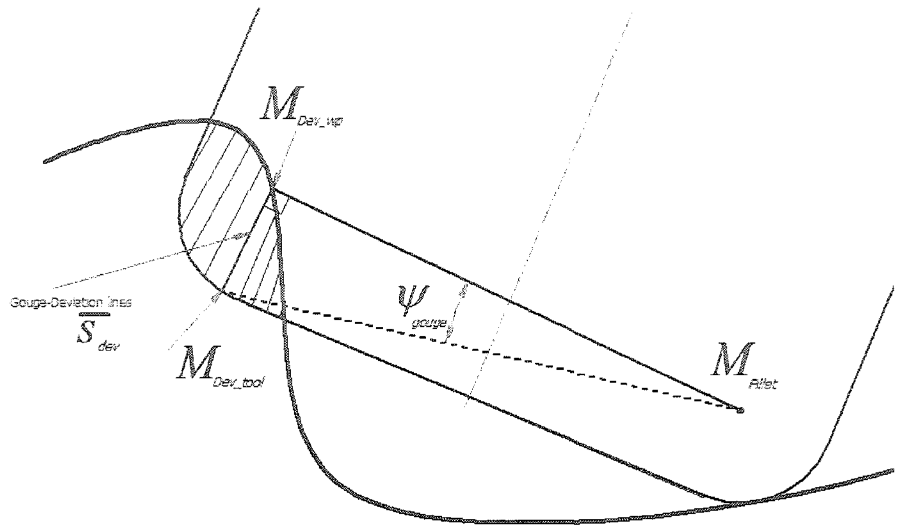


Figure 6.5: Maximum gouge distance and angle

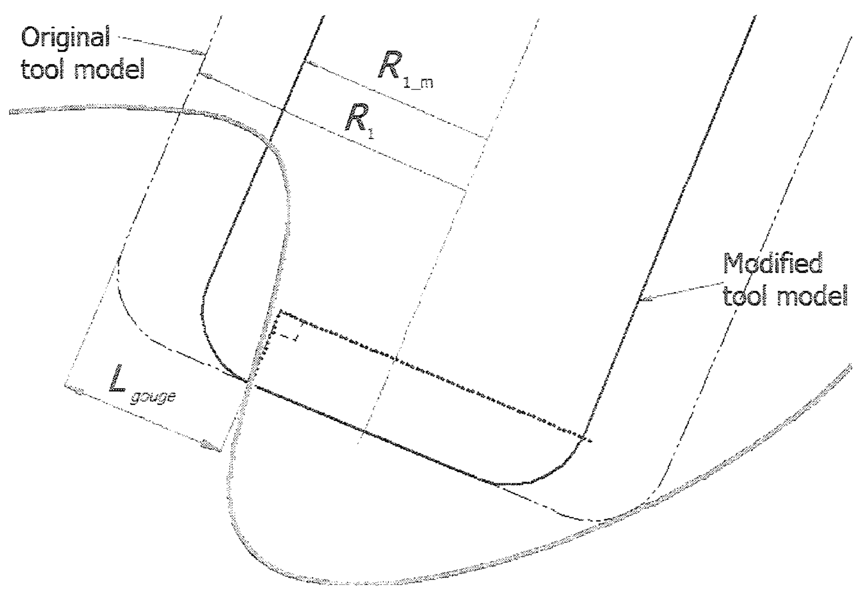


Figure 6.6: Tool model modification (collision compensation)

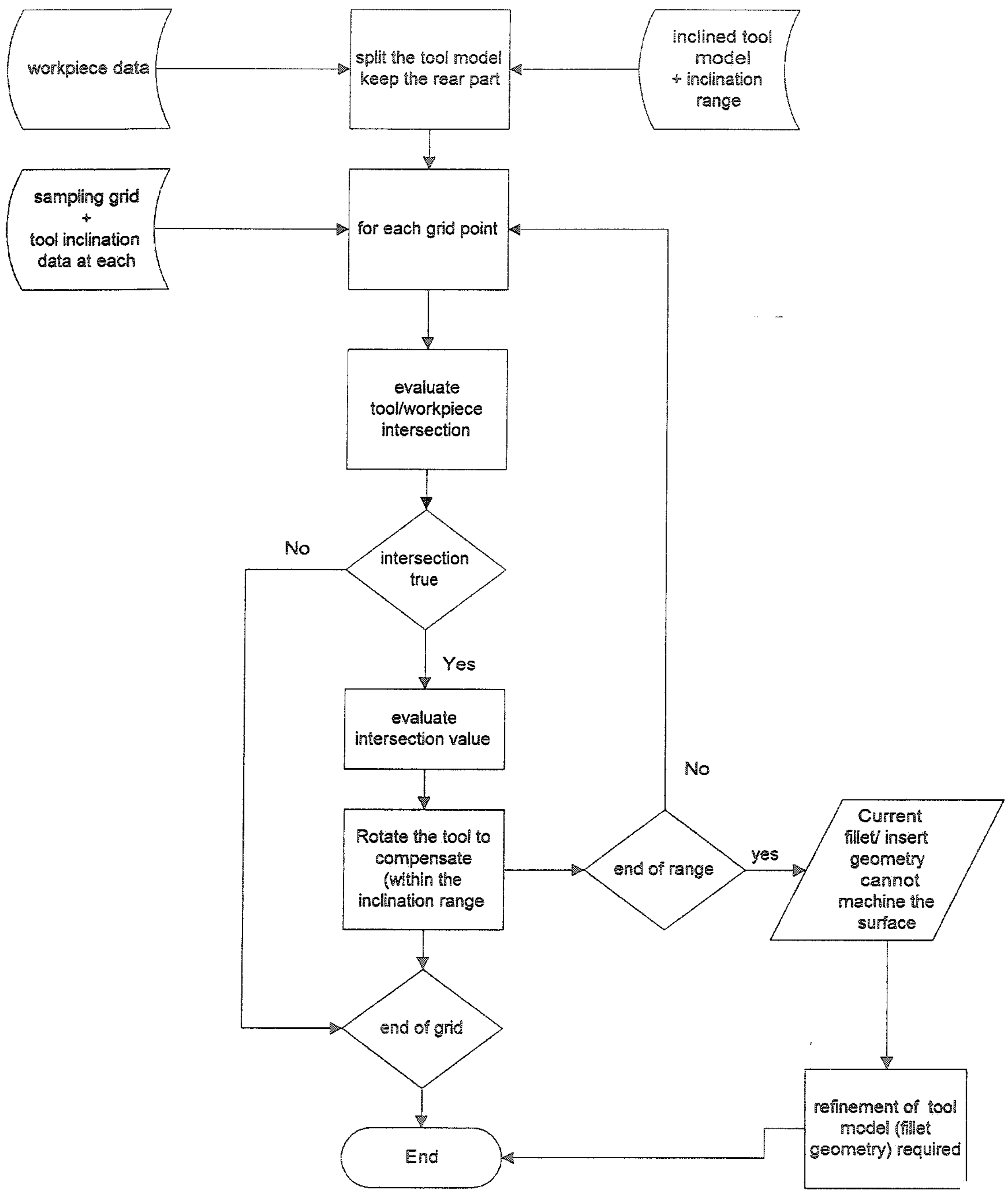


Figure 6.7: Rear gouge evaluation algorithm

6.3.2 Tool Shank Collision Evaluation

After sorting-out tool positioning and rear gouge problems, the tool path optimization procedures reach another stage. The combined tool movements may create a situation when the tool cannot reach certain parts of the workpiece surface. This is due to the interference of the tool shank with the workpiece preventing the tool from reaching the desired contact point. This situation needs to be evaluated and solved before generating the tool path. Leaving this problem unsolved, would prevent the compilation of the cutting path or even cause machine tool damage.

To address this problem, the developed algorithm uses the tool-model shank part to perform interference checks at each tool path point. On detecting interference, the tool path generation process stops to sort-out the collision problem. The relevant tool part is evaluated against the semi-finished tool to get the amount of the collision.

A parting plane is used to create a section containing the tool/ workpiece profile. The magnitude of the maximum interference and the direction for compensation is then evaluated by maximum collision algorithm.

The maximum collision is evaluated in a similar manner as the maximum gouge problem. As shown in figure 6.8, the collision deviation vector is created as a normal vector to the line the connecting evaluation point M_{Coll_wp} reaching to the tool-section fillet center point M_{fillet}. This vector is defined in equation 6.7.

$$\overrightarrow{S}_{Coll} \times \overrightarrow{S}(M_{fillet}, M_{Coll\_wp}) = 0 \qquad (6.7)$$

The maximum value of the collision-deviation vector indicates the amount of tool shank gouge. The collision compensation algorithm is then called using the maximum collision-deviation vector.

To eliminate collision situation the tool inclination is varied. The value of the collision-compensation angle is obtained from equation 6.8.

$$\psi_{Coll} = \tan^{-1}\left(\frac{\left|\overrightarrow{S}_{Coll}\right|}{|S(M_{fillet} - M_{Coll\_wp}|}\right) \tag{6.8}$$

Before applying the required transformation to eliminate the collision situation, the compensation angle ψ_{Coll} must be checked to ensure that it is within the given feasible inclination range retrieved from the effective tool profile evolution phase $\psi_s \leq \psi_{Coll} \leq \psi_e$.

If the compensation angle was found to be out of range, tool model modification is then requested. The tool shank angle ϕ_{flank} needs to be altered by a value equal or greater than the compensation angle to solve this collision situation as shown in figure 6.9.

These conditions in equation 6.9, becomes a constraints for the system to satisfy before tool path generation is carried out. The outline of the collision detection and compensation algorithm is shown in figure 6.10.

$$\Psi_{tool} = \begin{cases} \phi_{flank} \leq \psi_{Coll} \\ \psi_s \leq \psi_{Coll} \leq \psi_e \end{cases} \tag{6.9}$$

At the end of all evaluations the other considerations, such as contact point and gouge, are rechecked to ensure that one variation in tool posture will not affect other requirements.

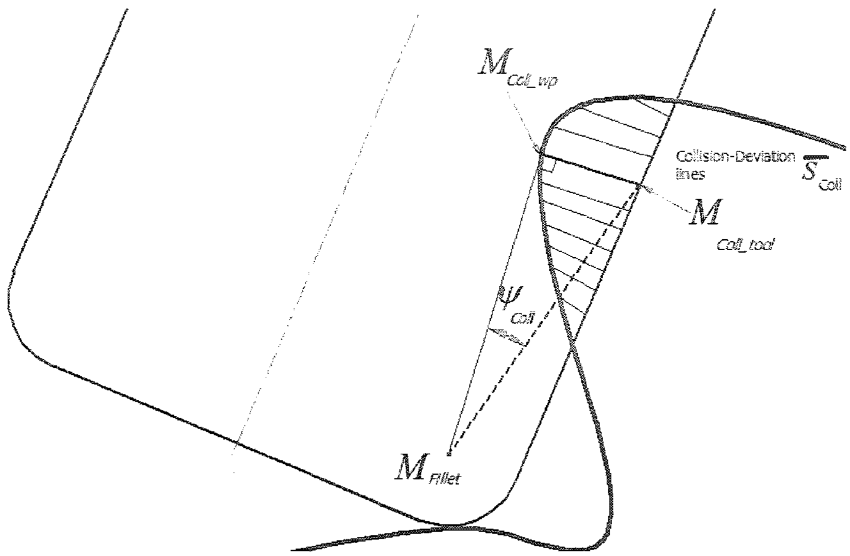


Figure 6.8: Maximum collision distance and angle

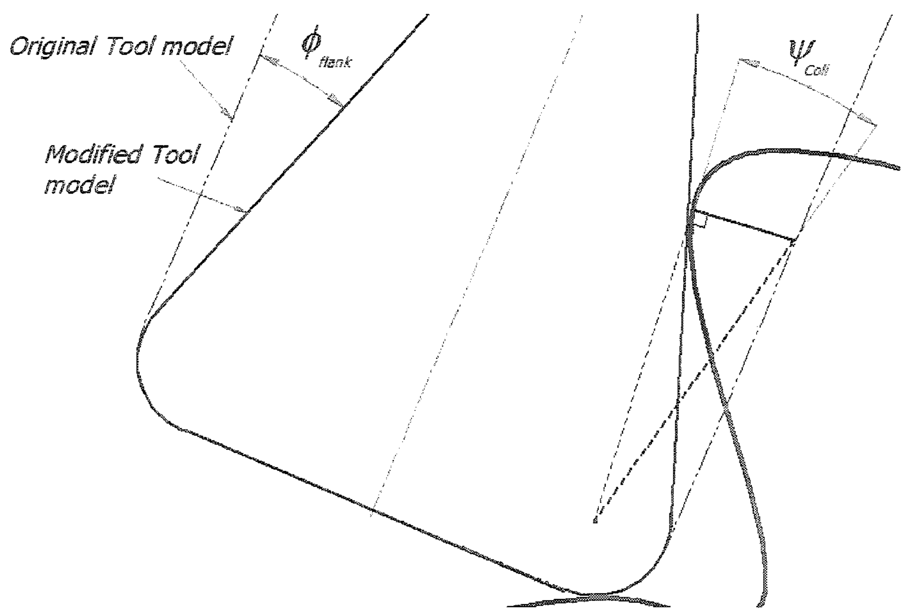


Figure 6.9: Tool model modification (collision compensation)

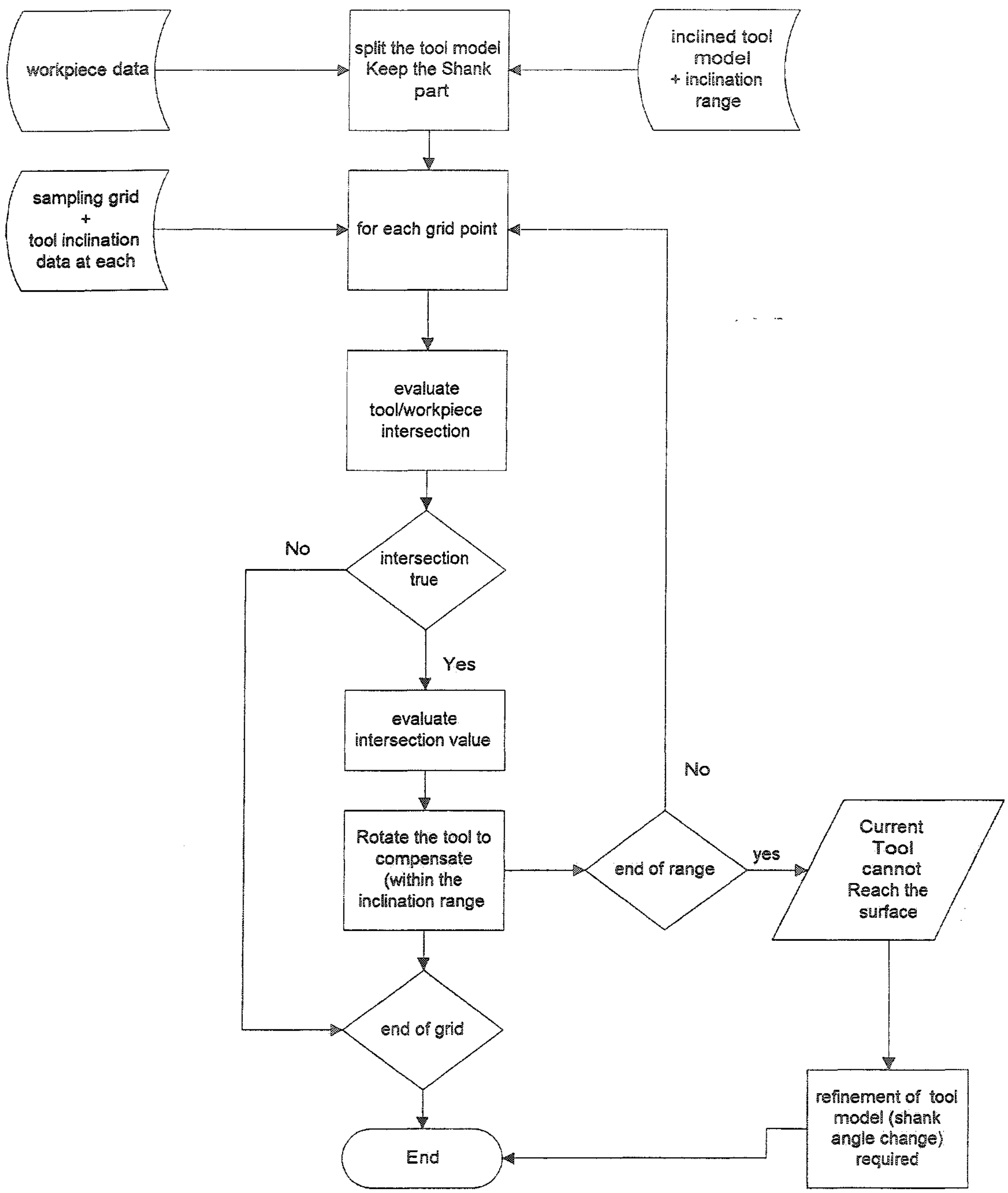


Figure 6.10: Collision evaluation algorithm

6.4 Tool path Output

On successful completion of all the optimization phases, the tool path generation process can now conclude. The locations of the tool tip and tool orientations at each tool path point is saved. A cutter-location output file (CL-Data file) is then generated to store the Cartesian coordinate and the orientation angles required to process the machining operation.

The CL-Data is then sent to machine tool post-processor to translate it to the machine tool language. To complete the CL-Data file, the cutting conditions (feed/spindle speed) must be included in the file. This data is obtained from the tool selection module by picking up the tool and recommended cutting conditions according to the workpiece material and the required surface finish.

6.5 Summary

In 5-axis milling, a valid tool path is not necessarily a usable one. Due to the complex geometries of the workpiece, the tool may accidentally cut into the finished part, in other cases the tool may hit unfinished areas of the part.

To avoid such situation tool gouge and collision detection and avoidance strategies were implemented. The evaluation of the gouge and collision is performed during the path planning phase and contribute in the refinement of the cutting tool geometry. After successfully finishing the tool positioning, gouge and collision evaluation, cutter location data file is generated. The remaining step before completing the tool path generation, is including the cutting conditions into the output file. This data should be obtained from the workpiece material and the required workpiece finish. Before performing the machining operation the tool path data should be translated according to the machine tool instruction using post-processor program.

CHAPTER 7
TOOL PATH PLANNING FOR FLANK MILLING

7.1 Introduction

The research in the field of multi-axis tool path optimization is not exclusive on utilizing the tool fillet/insert in generating the machined surface. While the majority of the work in five-axis milling is focused on point milling, there are more productive methods to achieve some tasks.

Flank milling can provide higher productivity than point milling for a certain class of surfaces. This surface class is ruled surface. Most of blades, turbines, and fans surfaces fall into the category of flank milling-enabled applications. This method utilizes greater area of the tool surface, the tool shaft or side surface, resulting in higher metal removal rate per machining pass. The major limitation of this technique is that it is exclusive for certain class of surfaces. For other surface types, it is very difficult and maybe sometimes impossible to use flank milling process in its manufacturing. Another limitation on the flank milling process is the cutting forces. Since the tool engagement in flank milling is much greater than the corresponding point milling operation, the components of the cutting forces are much higher. This effect complicates the process stability evaluations(e.g. Budak (2000), and Larue and Anselmetti (2003)).

The ruled surfaces best suitable for flank milling is the developable surfaces with minimum twist between the directrices. Non-developable and twisted surface can still be machined with this method, but with certain surface error.

The importance of this phase in the tool path optimization of five-axis machining is to include the different five axis milling strategies. The integration of this part with the rest of the system broadens the scope of the system and makes it more flexible in dealing with different classes of free-form surface.

In this work developable ruled surfaces were investigated. Straight flank cutters were used to investigate the tool positioning errors. The investigations included the surface errors resulting from the relative position between the tool and workpiece, and the errors resulting from tool shape and tool path-segments approximations. These errors were evaluated, and the tool path forward step was optimized to reduce the errors.

Figure 7.1, represents a general layout of the developed algorithm, more detailed algorithms can be found in appendix A.5.

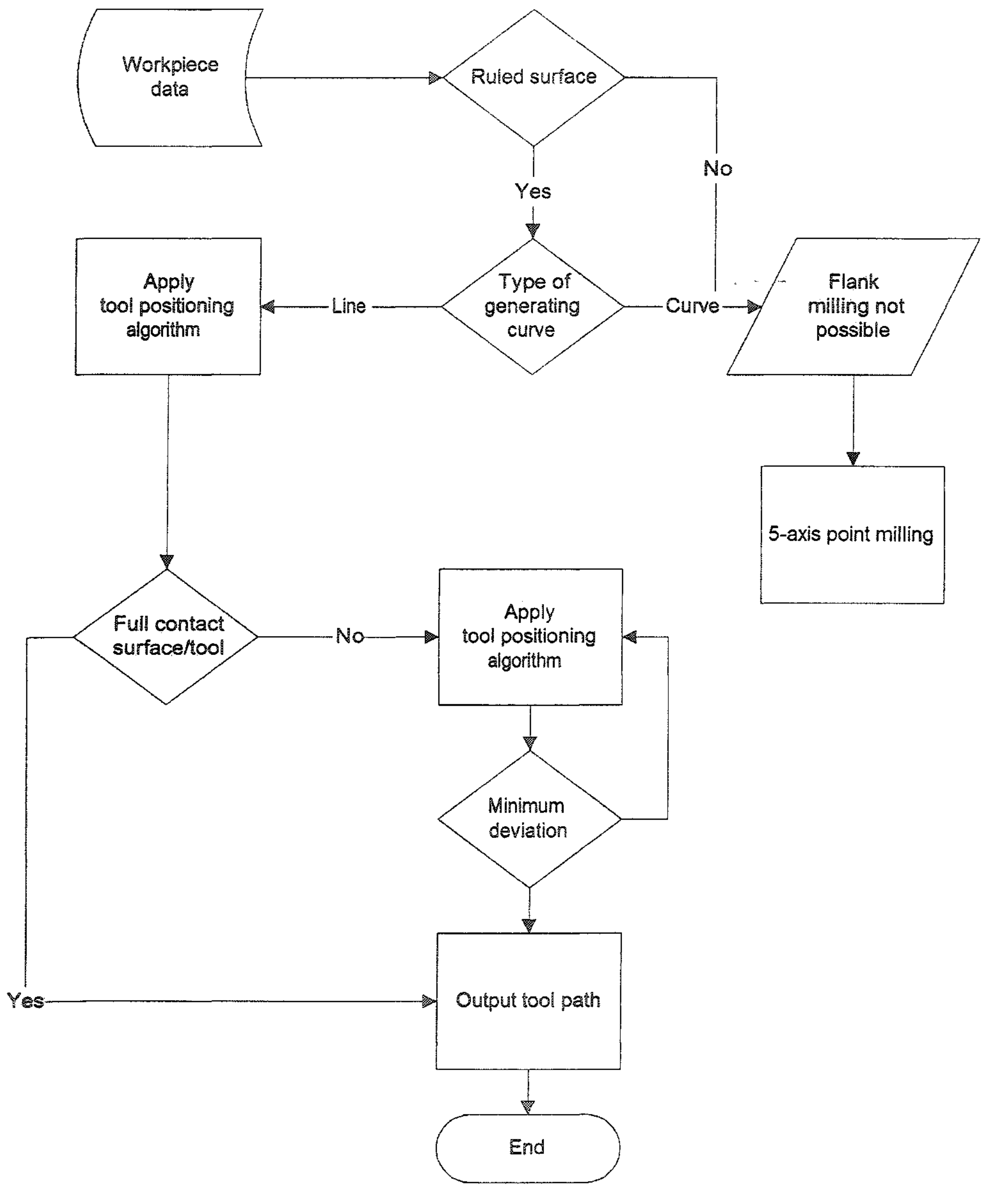


Figure 7.1: Flank milling algorithm

7.2 Extraction of Tool Data and Tool Model Creation

Before the tool placement evaluation, the tool model must be known. As a part of the tool path optimization process the workpiece surface is evaluated to extract its topological features. In a similar methodology to the evaluation of surface topology as discussed in chapter 3, the evaluation of curvatures of the surface curves was the focus of this part.

In point milling module the normal surface-curvatures were evaluated to extract the data needed for initial tool model creation. In flank milling, since the tool positioning strategy is different, another approach was taken.

The curvature evaluation is limited to the surface directrices. Minimum radius of curvature of the concave parts of the curve, based on the normal vector direction as shown in figure 7.2, was used to constraint the maximum tool diameter as described in equation 7.1. This ensures that the tool will fit inside the surface profile and surface errors due to tool size will be eliminated.

$$R_1 \leq \frac{1}{k_{\max}}, R_m = R_1 \tag{7.1}$$

The second tool parameter is the tool length. Evaluation of the oriented workpiece bounding box gives an indication for minimum tool length required to machine the surface in one path.

These two parameters are sufficient to create cylindrical tool model. The tool model creation module developed for this work (see chapter 4), was used to create the tool model. This model will be used in tool path generation and error checking in next sections.

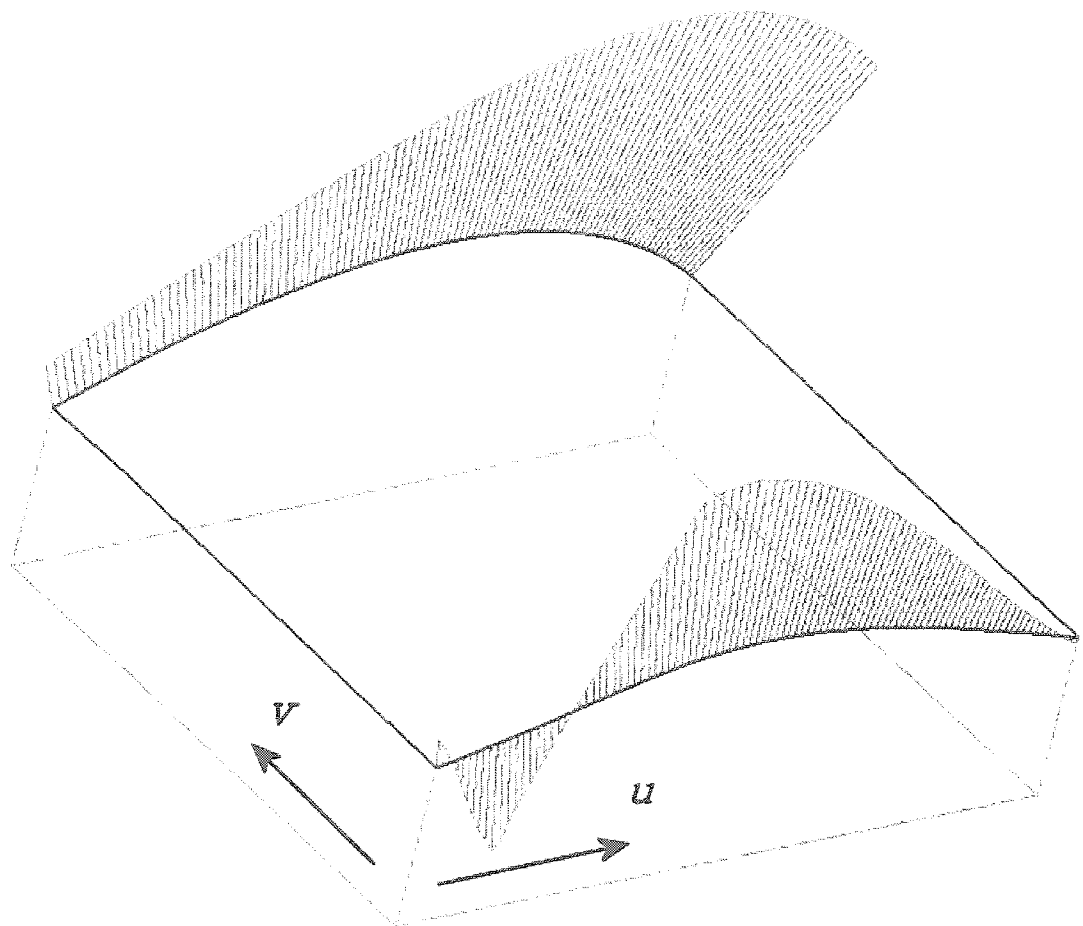


Figure 7.2: Curvature analysis of ruled surfaces

7.3 Flank Positioning of Cylindrical Tools

In flank milling of a ruled surface, two tool placement strategies are commonly used. In the first strategy, the tool is to be positioned tangent to the ruled surface directrices. The second strategy, is to position the tool tangent to the rule lines. For the developable ruled surface these two strategies converges into one, since the ruled surface is generated from straight lines. This means that by positioning the tool tangent to the rule line, it will be tangent to the directrices as well.

After the selection of the tool placement strategy, the locations of the tool evaluation positions must be determined next. The evaluation of tool movement can be carried out in many different ways. Some research has used the directrices curve to create an offset curve over the workpiece at a distance equals the tool radius. A swept volume of the tool profile is then used to create the tool path. Another method is to approximate the tool forward movement into straight lines.

In this work tool path approximation was used. The forward step was controlled by the surface topology. The deviation between the design surface and machined surface was calculated. The maximum deviation was compared to surface tolerance, and optimal forward step at each side of the surface was chosen.

As a start point, the tool path evaluation begins from the workpiece surface parameterizations, following the same iso-parametric evaluation used in chapter 3.

The forward step was controlled by the evaluation of chord length deviation between the end curves and the tool path segment. Due to the nature of the parametric surfaces, the number of parameters used to generate the surface must be the same at both ends. Meanwhile in the Cartesian space, the separation distance between two parametric distance is not necessarily equal at both ends of the surface. In the case that the surface is wider at one end than the other in the Cartesian domain, the forward step must be evaluated at both end curves.

The evaluation of forward step in the Cartesian domain at both ends, ensures that the tool path step will be uniform. This controls the error at both ends of the surface.

The tool is placed using the normal vectors at each tool path evaluation position. The tool axis is placed in a position along the normal vectors at the start and end of the surface as shown in figure 7.3. This position is at a distance equals to the tool radius.

The verification of the tool path is carried out by solid instantiation technique as shown in figure 7.4. The updated workpiece verifies the final shape of the machined surface.

The surface errors along the end curves is then evaluated, and the deviation calculations correct the tool path segments length accordingly. The error along the rule line is then evaluated and saved for further investigation

In the following sections, error evaluation and compensation strategies will be discussed.

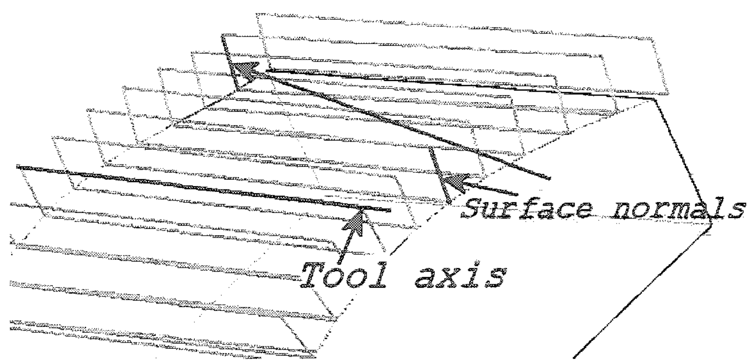


Figure 7.3: Tool axis placement relative to workpiece surface

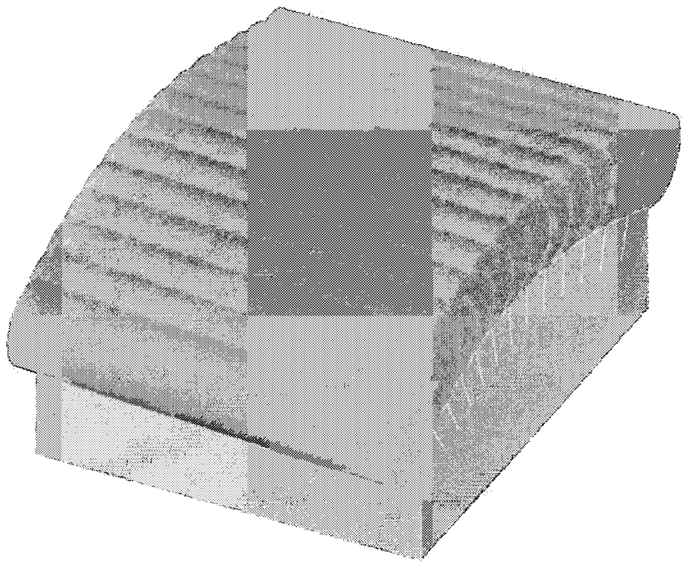


Figure 7.4: Solid tool placement along tool path

7.4 Surface Deviations Detection

During the tool positioning and tool path generation, surface errors should be detected and eliminated to ensure the validity of the tool path. In this work two error sources were evaluated. The deviation from the surface along the end curves of workpiece surface, and the interference of the tool with the workpiece along the tool axis. These errors result from tool positioning, and their values were evaluated.

The deviations along each of the end curves were evaluated separately. The span between each two parametric evaluation points was checked. If the deviation from the generating curve is higher than the surface tolerance, span reduction algorithm was then applied. The reduced span second point was then used as start point for the next span evaluation. This strategy was used to evaluate both end curves. The minimum of the two evaluation results was then used as the forward step. Figure 7.5, illustrates the deviation difference when calculating the forward steps.

The second error source in flank path planning is the deviation between the

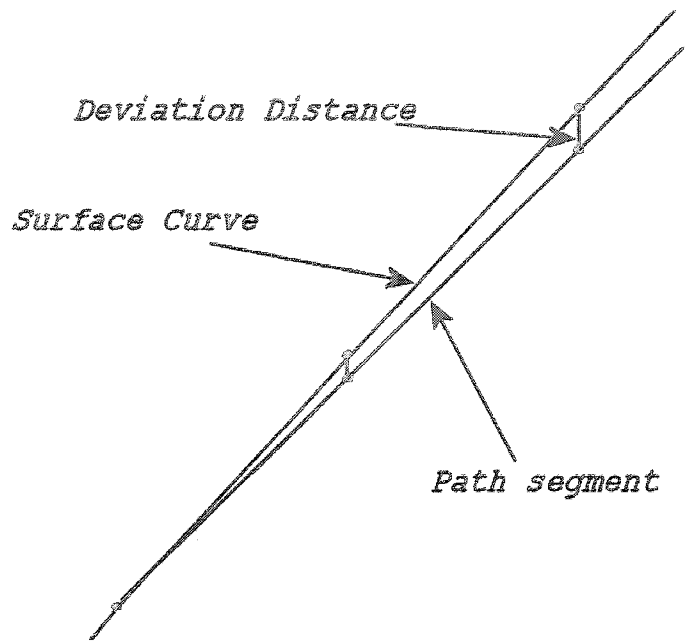


Figure 7.5: Deviation evaluation along directrices

workpiece and tool along the tool axis. This deviation produces over-cuts in the workpiece surface. This error was evaluated by the checking of tool/ workpiece interference. The existence of interference indicates surface error existence.

The evaluation of the magnitude and distribution of the surface error is carried out by intersecting a plane passing through the tool axis and tool/workpiece contact curve as shown in figure 7.6. The distance between the tool intersection curve and workpiece intersection curve was evaluated along the tool length, this indicates the error distribution.

After the finalization of the tool positions, CL-Data file is generated, including the tool tip positions and tool axis orientation. The generated tool path together with the tool data and workpiece material is then sent to the machine tool post-processor to generate the cutting path.

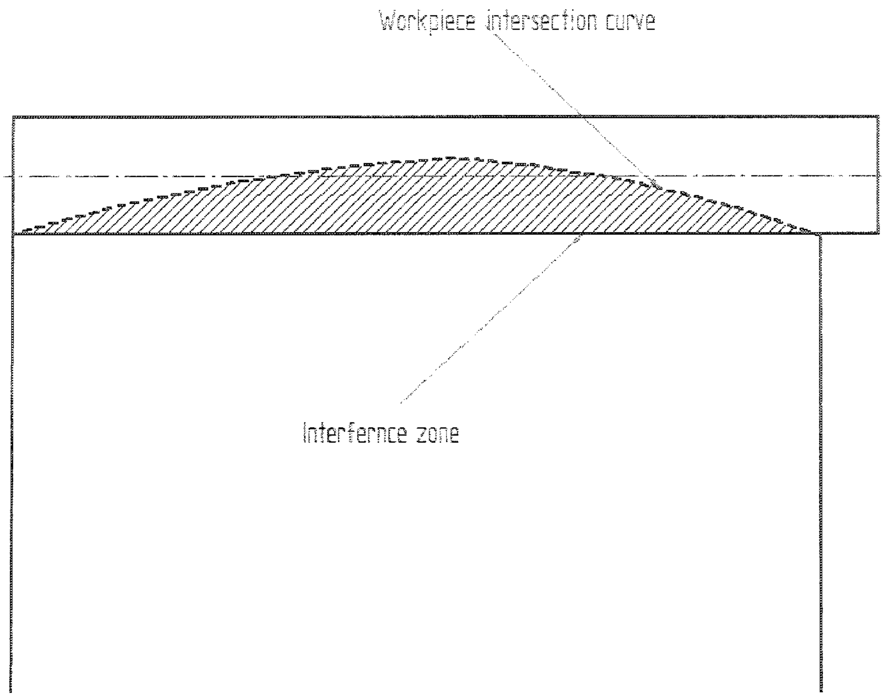


Figure 7.6: Deviation evaluation along tool/workpice contact line

7.5 Summary

The flank milling method always yields higher metal removal rates. Furthermore it can be used to overcome the chatter problem encountered in high speed point milling. On the other hand, certain characteristics are required for a workpiece to qualify for this method. Typically this method gives best results with developable ruled surfaces, especially low-twist ruled surfaces created by straight lines. In this work the investigation of the feasibility of the workpiece to be produced using this method was explored. Tool positioning and checking strategies were implemented to generate and verify the cutting tool path.

CHAPTER 8 MULTI-STAGE TOOL SELECTION

8.1 Introduction

The tool selection is an important task that can greatly alter the path planning strategy. For instance the choice of ball end mill can lead to simpler tool positioning strategy, but on the other hand leads to denser CL−Data file and/or larger cusps. Most of the available CAM systems start the manufacturing operation sequence by tool selection. The tool selection is achieved through user interaction and depends on his/her knowledge. Another feature in most of the CAM systems, is that the tool path checking is performed after the tool path generation, which requires the re-evaluation of the whole tool path. The need for special tools, especially for free form surfaces machining, initiated the need for new guidelines in tool selection in order to acquire the needed tool.

8.2 Tool Model Refinements

This phase is not a separate implementation, the tool model was re-created at least one time during the tool path optimization process. During the discussion of the different stages of this work, the tool model was mentioned in different contexts.

In the curvature evaluation phase (Chapter 3), the tool initial parameters were synthesized from the workpiece topology. The initial tool diameter , fillet radius, and tool length were determined based on workpiece curvatures and dimensions.

The initial tool model is then created based on the given constraints (Chapter 4). This model was used in all the following phases to evaluate and modify the tool path.

In the effective tool profile phase (Chapter 5), the tool model was evaluated against the workpiece profile. Then the tool inclination was used to vary the tool profile trying to find the optimal profile to eliminate surface errors. Tool diameter reduction maybe required to continue the tool path planning process.

In the gouge evaluation phase (Chapter 6), the tool rear part may touch the finished workpiece. After exhausting all tool inclinations, tool diameter change must be performed. The tool model refinement must be carried out before proceeding to the next phase.

In the collision evaluation phase, the tool flank and holder may collide with the workpiece. To avoid this situation, the tool inclination is altered. If the collision cannot be eliminated, then tool flank inclination must be changed before the tool path generation process can proceed.

During this discussion, the tool model could have changed at least one time. This multi-stage tool geometry refinement process reduces the machined surface error and enhances the tool path optimization process.

To finalize the tool path generation process the tool selection process must go through another final refinement step. This step will be discussed in the following section.

The developed system overview is illustrated in figure 8.1. Other phases of this work covers the tool selection procedures as well.

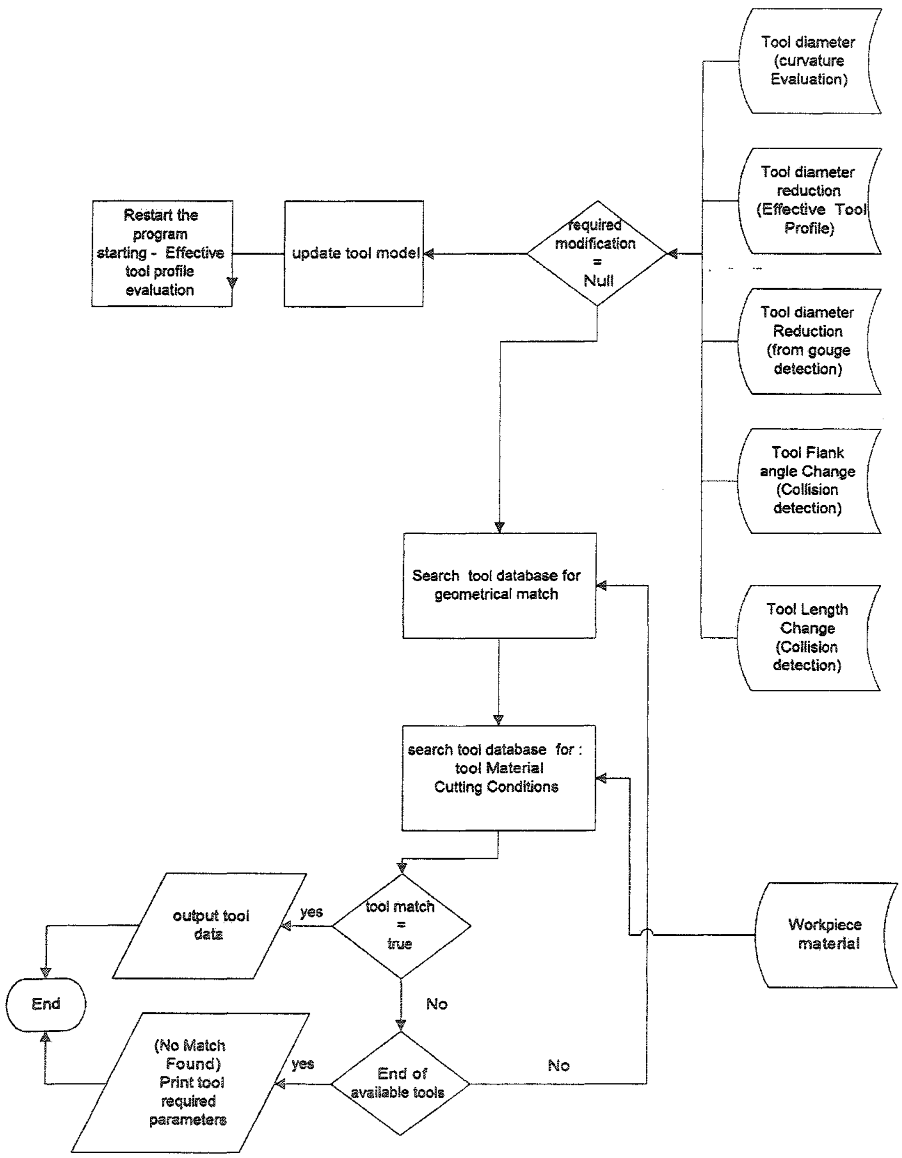


Figure 8.1: Tool selection process

8.3 Process Parameters Selection

This process must be carried out to finalize the tool and cutting condition. Using the workpiece material and the required surface finish as guidelines, the tool parameters was then selected.

Tool catalog, or computerized tool database can be searched to find a tool match based on the tool workpiece material and then tool geometry. If a tool match was found, the cutting process parameters can be selected or calculated. Other process optimization modules (e.g. Cutting forces optimization, feed scheduling, thermal analysis) are to be used to enhance the milling process optimization by contributing in process parameters selection and in tool path modifications.

The tool geometry resulting from this multi-stage process, may not match standardized tool geometry. Thus, these geometrical parameters may be used for the manufacturing of special tool.

After the selection of the new tool and finalizing the process parameters, the output tool path can be put into use, minimizing the need for tool path re-evaluation due to machining errors.

8.4 Summary

The developed tool selection algorithm is based on multi-stage refinement from the different tool path optimization and checking phases. This approach integrates the workpiece characteristics and the tool path requirement with final tool geometry. The developed system guides user judgment to select the tool.

CHAPTER 9
METHODOLOGY VALIDATION

9.1 Introduction

In the previous chapters, a detailed investigation of the different phases constituting this work was presented. The output from the developed work is an optimized tool path in the form of CL-Data file. In order to check this output, the developed system has error checking algorithms that report the errors and initiate modification algorithms. The developed system also used solid instantiation algorithms to update the semi–finished part model and visualize the tool movements.

In this chapter, experimental verification for the developed system output is presented. The main purpose of this experimental work is to verify system output. This include verification cuts using five axis machine tool, the measurement of cusp height, and the checking of the smooth transition of tool orientations during tool path progression.

A comparison study is also presented, between the output from this work and the output from a commercial CAM package. This comparison is carried out to elaborate the main differences between the tool path generation process in both cases.

9.2 Experimental Procedures

In order to verify the developed work output, verification cuts were performed. A MAKINO-MC56-5XA, five axis horizontal machining center with FANUC controller was used to perform these tests.

In order to minimize the effect of the other process parameters (e.g. cutting forces, tool wear, thermal effects, etc.), the workpiece material chosen for these tests was paraffine wax.

The CL-Data file (system output), will not be ready for use in the machine tool controller before post-processing. Thus, a post-processor was adapted for the needed conversion. This post-processer was adapted from the work by Tounsi and Elbestawi (2003a), based on inverse kinematic evaluations found in She and Lee (2000).

In these tests number of verification cuts were performed. The workpiece model used for this experimental work is shown in figure 3.2(b). This workpiece was used to represent various topological features which requires tool path checking and refinements.

The developed system was evaluated using the inputs shown in table 9.1. The surface error tolerance was altered from 0.5 to 0.1 $mm.$, and the required cusp limit was varied from 0.14 to 0.5 $mm.$, in order to test various system outputs.

Exp. No.	Part Tolerance ($mm.$)	Cusp height (mm)
1	0.5	0.14
2	0.1	0.3
3	0.1	0.5

Table 9.1: System parameters used for experimental tool path generation

The dimensional measurement was carried out using coordinate measuring machine (CMM). A ZEISS CMM was used to evaluate different profiles on each workpiece surface. The locations of the profiles were chosen to demonstrate the critical sections of the machined surface.

A second measurement was taken along a tool path direction. The reason is to track the cutter progression and to verify that gouge situations were resolved by the developed system before the tool path processing.

In the following section, the results of these experiments will be presented as well as the analysis and discussion of the results.

9.3 Experimental Results

The experiential procedures discussed in the previous section aimed toward checking the validation of the theoretical concepts developed in this work as well as the results reported by the system outputs.

This experimental investigation focused on proving the following developed system output characteristics;

1. The consistency of the actual cutter movement with the simulated results.
2. The elimination of surface errors due to gouges and collisions.
3. The smoothness of transitions in tool orientation during tool path progression.
4. The cusp height required was obtained by the actual cuts.

The first characteristic was proved by the processing of the developed system output by the machine tool. The tool path movements produced uniform cutting paths on the workpiece surface. The machined profile segments were consistent with the simulated machined profile of the workpiece.

9.3.1 Cusp Height Evaluation

The developed system generates the tool path using cusp height value as one of its inputs. The system then tries to respect this limit throughout the tool path progression.

The machined surface was measured at different sections for each of the experiential cuts described in table 9.1. The results of these measurements are shown in figures 9.1, 9.2, and 9.3.

Results were automatically scaled, by the machine, to clearly demonstrate the cusp profile between two adjacent tool path. To simplify the comparison of the results, the summary of the measured results was listed in table 9.2.

Exp. no.	Required Cusp (*mm.*)	Section no.	Measured Cusp (*mm.*)	Difference (*mm.*)
1	0.14	1	0.115	-0.025
		2	0.184	+0.044
		3	0.18	+0.04
2	0.3	1	0.269	-0.031
		2	0.226	-0.074
		3	0.277	-0.023
3	0.5	1	0.113	-0.38
		2	0.201	-0.3
		3	0.56	+0.06

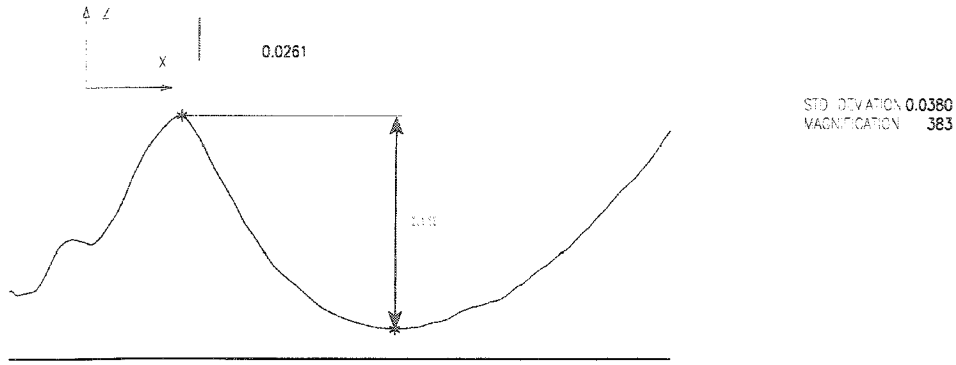
Table 9.2: Experimental results for cusp height measurement

From the summary of the results it can be seen that, for a given value of cusp height, the results were around below the required limit. It can also be seen that the measured cusp values shown in (Figures 9.1(a), 9.2(a), 9.2(b), 9.2(c), 9.3(a), and 9.3(b)), were less than the corresponding required cusp height. While the measured values shown in (Figures 9.1(b), 9.1(c), 9.3(c)), were higher than the required cusp height.

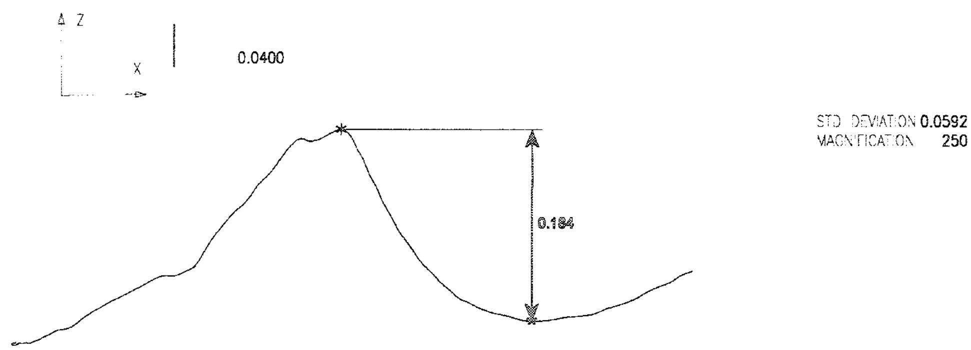
The deviation from the given cusp values can be attributed to several factors, among which;

- The nature of the workpiece free-form surface makes it impossible to match the effective tool profile at all its points. Although the developed system checks the tool placement at each tool path segment start and end, the tool orientation varies while moving from the start to the end of the segment.

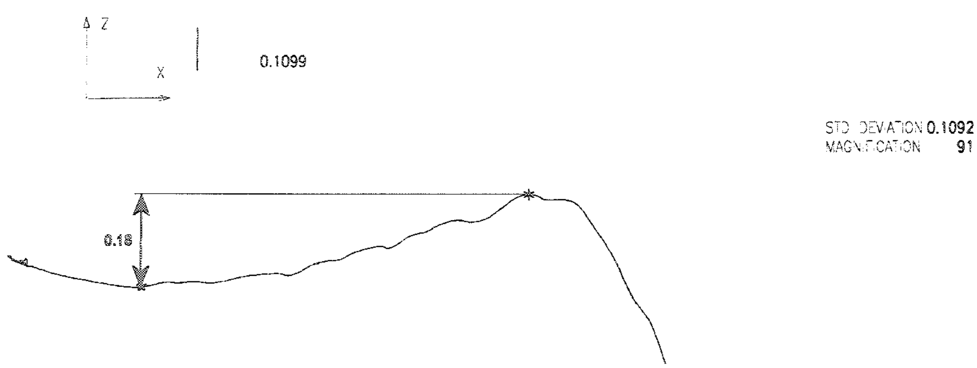
- The combined movements of the tool affect the shape and orientation of its effective tool profile, which is responsible for creating its negative imprints on the workpiece. Thus predicting the instantaneous tool profile throughout the tool path is required to overcome this problem. In addition to the other tool path checking algorithms, this may extend the evaluation time indefinitely.
- The deviation between the actual cutting tool profile and the tool model used in tool path generation and evaluations.
- Other sources of machine tool deviations, like vibrations and feed axis properties can contribute in creating these deviations.



(a)

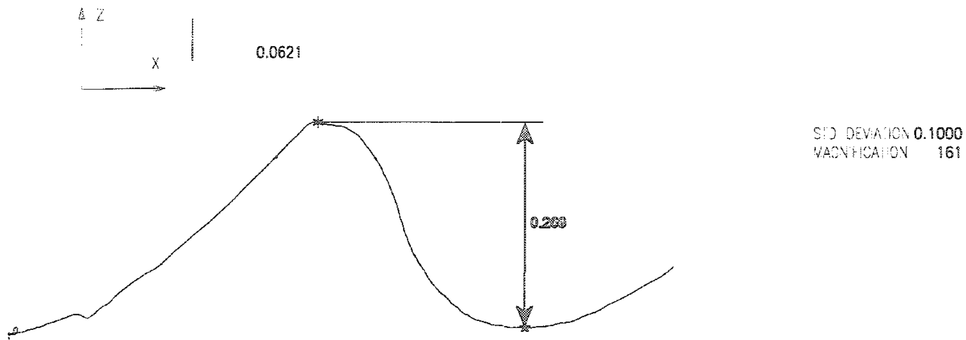


(b)

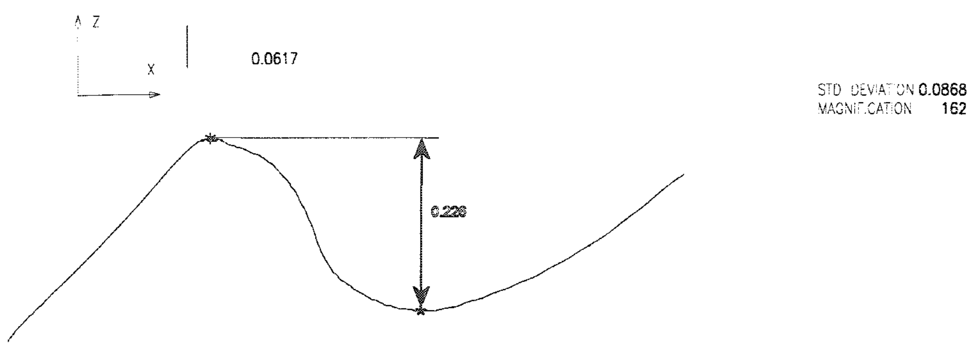


(c)

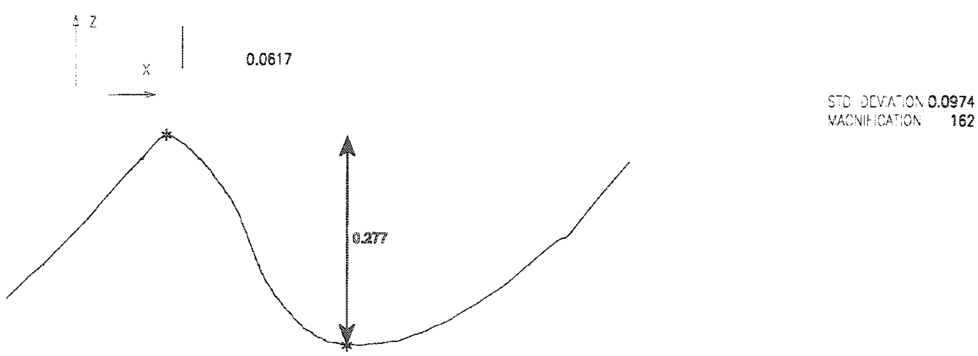
Figure 9.1: Profile measurement with tolerance=0.5 mm.,and cusp height=0.14 mm.



(a)

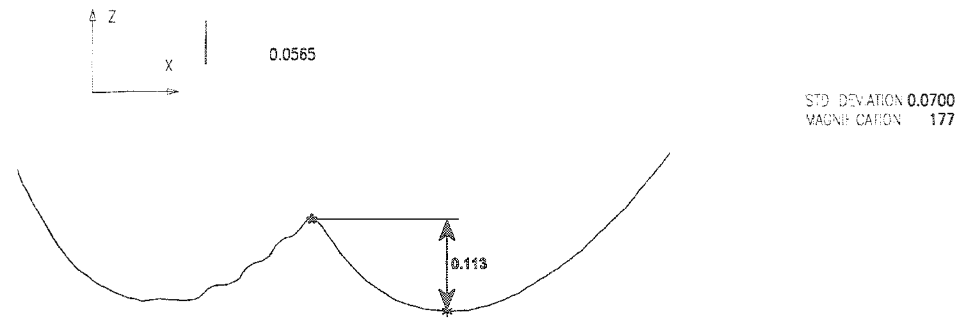


(b)

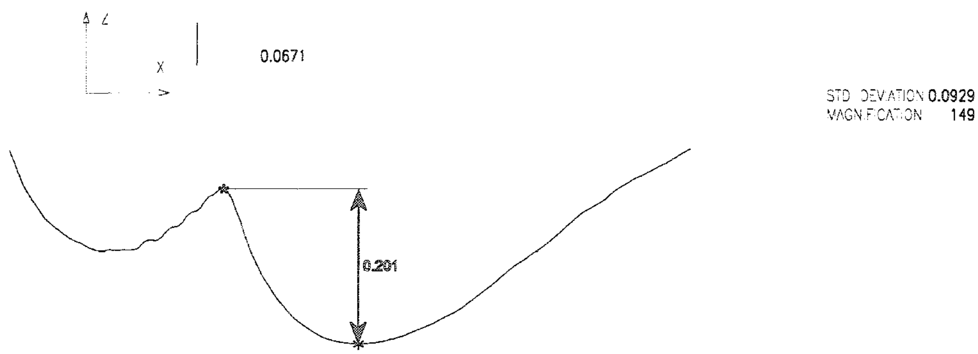


(c)

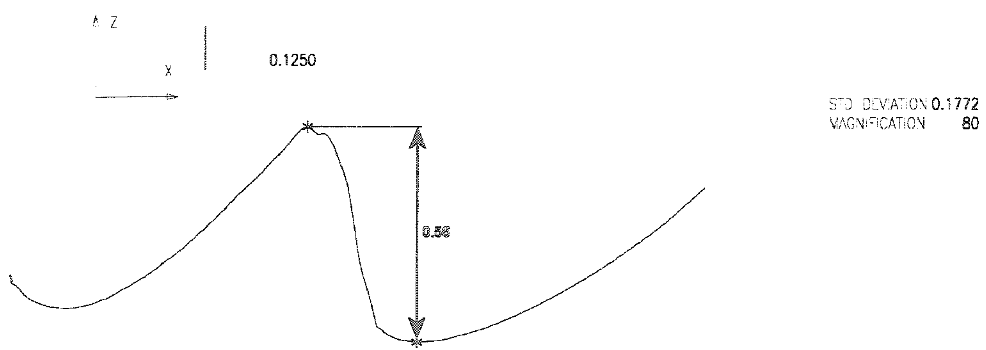
Figure 9.2: Profile measurement with tolerance=0.1 $mm.$,and cusp height=0.3 $mm.$



(a)



(b)



(c)

Figure 9.3: Profile measurement with tolerance=0.1 $mm.$,and cusp height=0.5 $mm.$

9.3.2 Tool Path Progression Evaluation

The developed system generates the tool path by approximation of the tool movements by straight lines. The length of these line segments are checked and corrected to ensure that the machined surface will be produced within the required part tolerance.

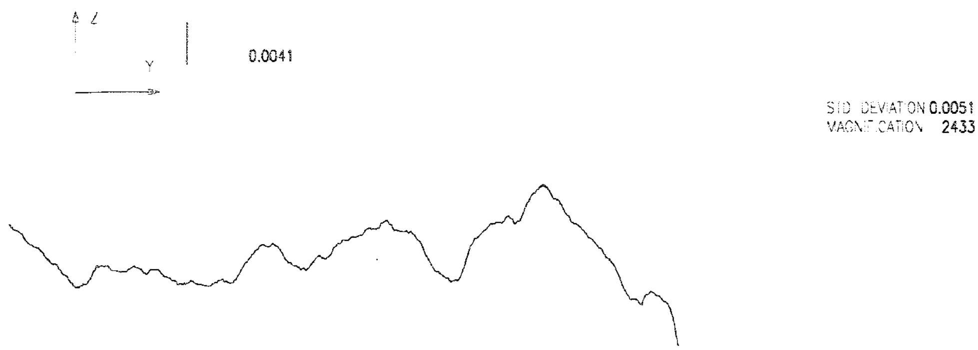
The machined surface was measured along the tool path to capture the traces of the tool movements along a single cutter path.

The measured results represent the cutter trace along 10 $mm.$ of a single tool path. This length was chosen to capture several consecutive tool path segments.

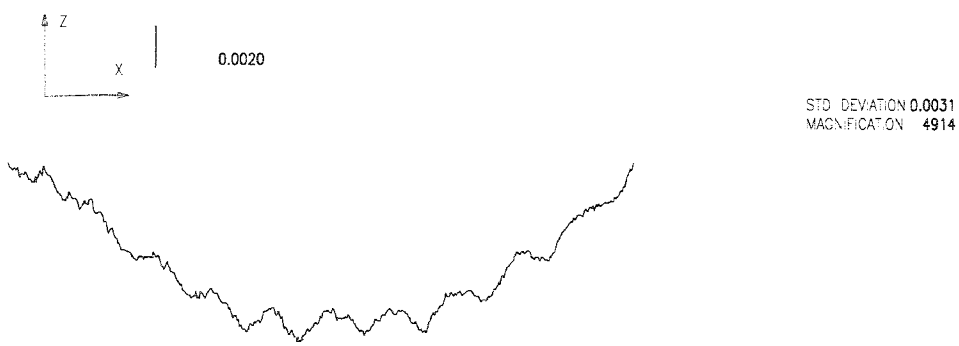
The results of these measurement are shown in figure 9.4. The figures were scaled to capture the surface profile at best resolution. These figures show the actual profile produced by the tool during single tool path.

It can be seen from the figures, that the fluctuations of the tool trace is very small ($\sim$ 0.008 $mm.$ as shown in figure 9.4(a)), ($\sim$ 0.002 $mm.$ as shown in figure 9.4(b)). These values are small given that the allowed surface error for these tool paths ranges from (0.1 to 0.5 $mm.$). The result from the third measurement (see figure 9.4(c)), resulted in very smooth profile.

The developed algorithm evaluates the tool path forward step by using surface tolerance value as an input. The deviation from the given tolerance values can be attributed to the conversion algorithms used to evaluate the tool path forward step. Since the evaluated workpiece surface is composed of space curves, the relationship between the tool path forward step and the surface deviation is non-linear. The developed system tries to reach the given tolerance. The evaluation may converge at smaller values than of the required tolerance, this value is used in tool path generation phase.



(a)



(b)

(c)

Figure 9.4: Forward step profile measurements

9.4 CAM System Comparison

In this section the output results from the developed system is presented in comparison to the output from CATIA version 5, CAM package as an example of state of the art commercial CAM software.

In this comparison the output tool path from CATIA was produced using different methods available in the software. These options are constant side step, constant cusp and optimized tool orientation. This output is then compared to the developed system output.

9.4.1 Case Study Comparisons

The process of tool path generation in CATIA, starts by selecting the surface to be machined. A default tool selection can be used, or the user can create/select a tool.

The comparison was carried-out using optimized tool selection recommended by the developed system. The used tool was 2 mm. diameter flat end mill cutter.

The output of the developed work was produced based on cusp height value of 0.14 mm. It can be seen from figure 9.6(a), that the optimized step-over distance was 0.95 mm. The cusp height produced in this area is much smaller than the specified cups value. Figure 9.6(b), represents another cross section at the highest curvature section of the workpiece. It can be seen that the maximum cusp height was 0.11 mm. This result is less than the required height, but it must be noted that the evaluation of this workpiece had taken much longer time than the same tool path generation process in CATIA (23 hours and 3 hours respectively).

The results for the same workpiece evaluation carried out on CATIA is shown in figure 9.7. This output was generated using different tool path criteria, to emphasize the difference between the ability of the developed system to select the tool path parameters and the CAM package user dependency.

Figure 9.5, shows the machined workpiece using 10 mm. diameter flat end mill cutter. It can be seen from the figure the gouge and undercut marks on the surface.

Figure 9.7(a), shows the updated workpiece model produced using CATIA. The process selected for this output was constant step-over distance of 1.0 mm. It can be seen from the figure that the actual cusp height was 1.28 mm..

Figure 9.7(b), shows the same updated workpiece model using cusp height of 0.1 mm. and fixed inclination angle of 5°. It can be seen from the figure that the actual cusp height was 0.905 mm.

The updated workpiece output shown in figure 9.7(c), was produced for the same required cusp height of 0.1 mm. but using optimized inclination angle. It can be seen that the actual cusp height was 0.327 mm.

From the figures, it was shown that the value of the actual cusp height with constant cusp method and with constant step-over method are higher than the desired value. This can be attributed to the approximations used by the CAM package to speed up the tool path evaluation.

In the course of this comparison some limitations were observed. In CATIA, the surface evaluation is carried-out by approximation of the free-form surface into facets. This speeds up the tool path evaluation, but on the other hand limits the accuracy of the evaluation. The minimum value that can be given to CATIA as machined surface error, and cusp height is 0.1 mm., otherwise the evaluation will not be completed. This creates a limit on how accurate your tool path evaluation can get. It was also observed that,the minimum value that can be given as input for step-over distance in CATIA is 1 mm. In the presented comparison the optimized step-over distance used was 0.95 mm., which means that the high cusps values produced in some part of CATIA output maybe partially attributed to this restriction.

The evaluation time for the developed system was about 23 hours, while it was 3 hours for the CAM package. This difference is directly related to the required accuracy of output.

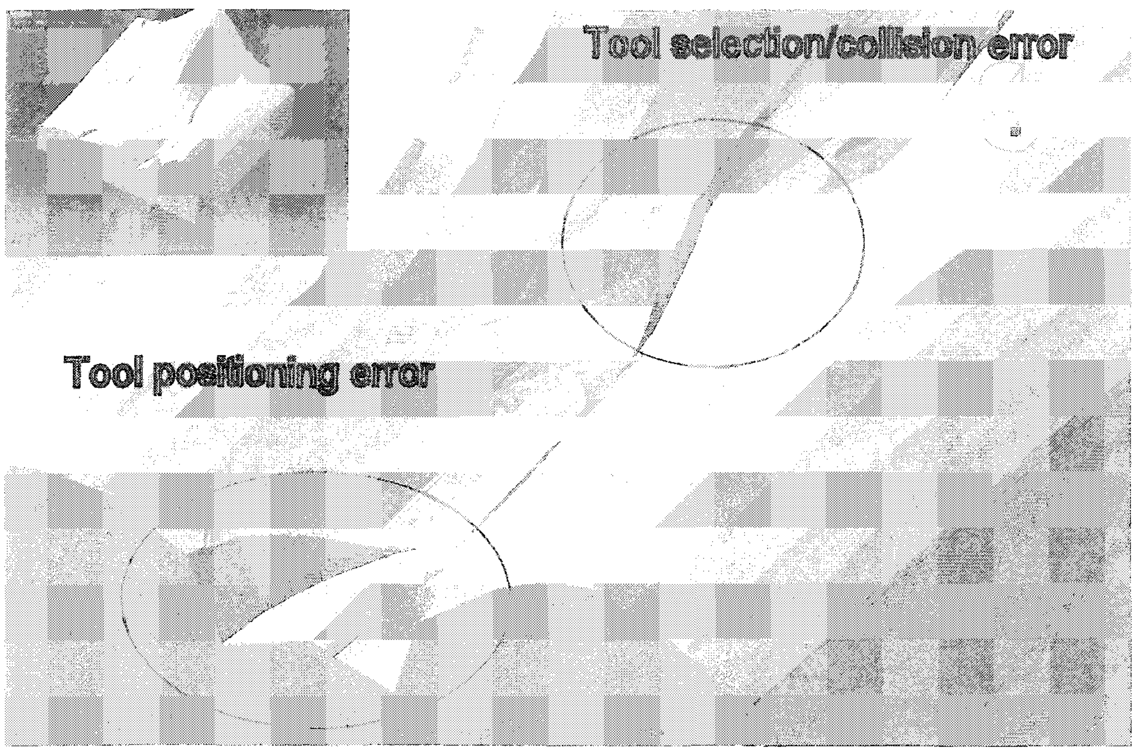
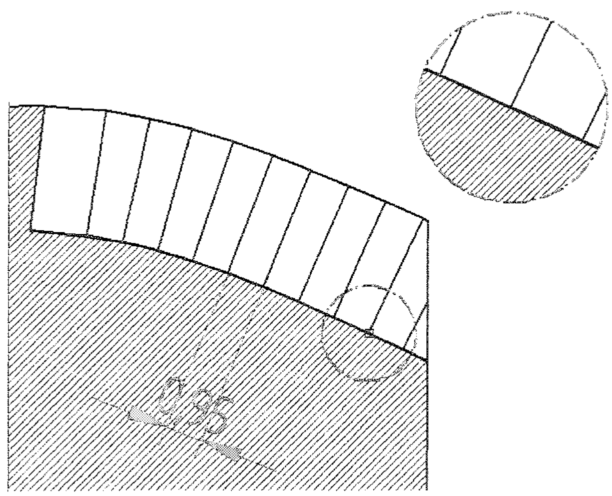
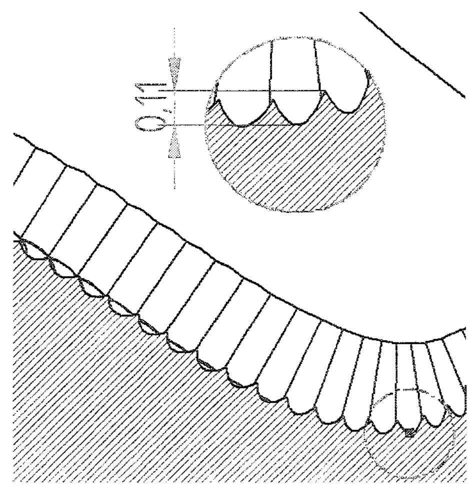


Figure 9.5: Workpiece output using flat end mill cutter with diameter = 10 mm. (optimized output)

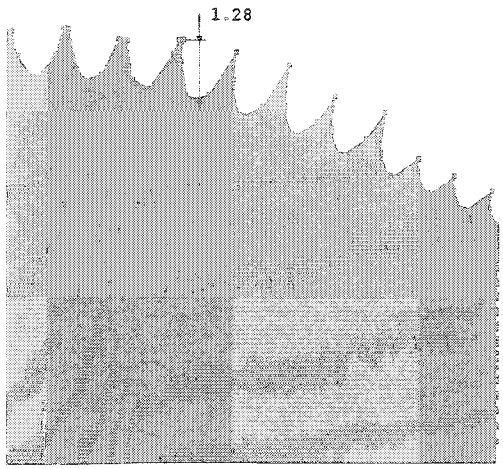


(a)

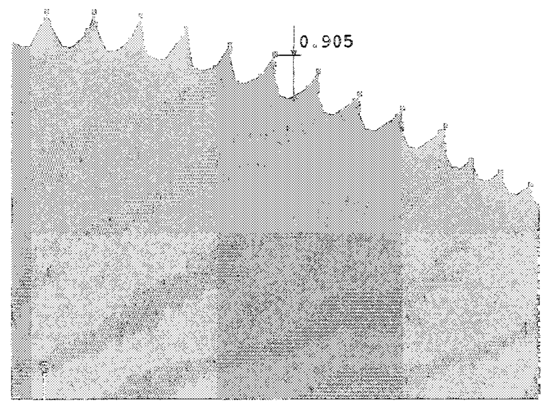


(b)

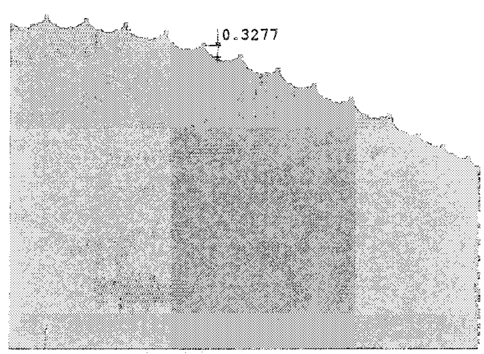
Figure 9.6: Developed system output using 2mm. diameter cutter 0.14 mm. cusp height(current work)



(a) Constant step-over distance= 1 mm.



(b) Constant cusp 0.1 mm. fixed inclination 5°



(c) Optimized tool orientation with cusp height 0.1 mm.

Figure 9.7: CATIA output using 2mm. diameter cutter (directional scale ratio =1)

9.5 Summary

The developed five axis tool path generation and optimization system results were verified. This verification was done using two methodologies. In the first methodology, an experimental procedures were developed to verify the validity of the developed system output.

The second verification was carried-out by comparing the developed system output to a commercial CAM package output to demonstrate the difference and clarify the significance of this work.

The experimental results showed that the developed system tool path output can be considered valid. The measurement of cusp height yield results that are satisfactory with minor deviations in some cases. This deviations can be attributed to several machining process characteristics.

The comparison between the commercial CAM package and the developed system showed the difference in the sequence and interactive user input. The developed system integrates tool selection and cusp height evaluation in the tool path generation and optimization process. It was also clear that the CAM package surface approximations led to faster evaluation, while resulting in surface errors and restrictions on the attainable output accuracy.

CHAPTER 10
CONCLUSIONS AND RECOMMENDATION

10.1 Introduction

The main objective of this work has been focused on the optimization of five axis milling process from the geometrical point of view. This work included tool path planning, gouge and collision avoidance, and tool selection. This chapter will summarize the conclusions of the work presented in this thesis, then the contributions of this work followed by the recommendation for future research.

10.2 Summary and Conclusion

Five axis milling is characterized by complex tool movements. The variation in cutter orientations requires more in-depth investigation in tool path generation and verification processes. The tool path validation process should take into account the various sources of errors. A solid model based on five axis tool path optimization was developed. In this work the geometrical aspects of the tool path errors were taken into account. This included tool placement checking, back gouge detection, and collision detection processes.

In most commercial CAM systems, the process of tool path generation is carried out in the following sequence; selecting the part to be machined, tool selection, tool path generation, and tool path verification. This means that the tool selection is ultimately a user dependent process.

In this work, an integrated tool selection process was developed. Different tool geometries were explored and the modification of this geometry was allowed to permit greater flexibility in reaching the optimal tool path. This process links the tool choice to the workpiece topology and to the machining process parameters. This ensures that the selected tool will not produce surface errors because of its shape interaction with the workpiece.

The approximation of the tool and workpiece models are always used in commercial CAM packages to speed up the calculation. The accuracy of this calculation and the precision of the output is greatly affected by this approximation. In this work a solid-model based system was developed. While the calculation time is greater than commercial CAM packages, the system can perform the tool path generation and verification calculation to much greater accuracy. This improvement is attributed to the use of the original workpiece solid model in the evaluation process.

In this work several evaluation algorithms and new evaluation concepts were developed. In addition, new tool model, and software were developed. The following are discussions of individual phases of this thesis work.

10.2.1 Curvature Based Tool Path Planning

Solid model based strategies were developed for the interrogation of the workpiece free-form surface topology. The following objectives were achieved in this phase;

- The original solid workpiece model was used in the evaluation instead of surface approximations.

- The developed surface interrogation strategies were based on surface parametrization. Using iso-parametric evaluation points, as a start point, leads to faster conversion towards the optimal locations of the tool path evaluation points, since these points were used in the creation of the workpiece model in the first place.
- Curvature evaluation strategy was developed using normal curvature of the surface as a measure to indicate the critical points on the workpiece surface.
- Initial tool geometry selection process was developed to determine the initial tool diameter, fillet diameter, and tool length. This process was based on the workpiece topology and geometry inspections.
- New tool path forward step evaluation strategy based on curvature change evaluations was developed. The curvature change was never used before in optimization of five axis machining process. This improves the accuracy of the interrogation algorithm in detecting the parts where the curvature interrogation evaluation overlooked. These parts were proven critical for tool path generation process.
- The developed algorithm checks for the surface deviation and ensures that surface errors are within the given part tolerances before allowing the next evaluation phase to start.

10.2.2 Generic Tool Model

A solid-model based generic tool model was developed. The properties of the solid model is essential in the tool path generation and optimization process. The following objectives were achieved in this phase;

- Solid modeling kernel algorithms were used to create the model. These algorithms takes advantage of the native modeling environment used to create the workpiece. This eliminates the need for mathematical or topological approximation needed for other systems to evaluate the cutting process (e.g. Tounsi et al. (2002), and Jensen et al. (2002a)).

- The developed model requires less number of parameters than the other models found in literature (e.g. Lee (1998b), Karunakaran et al. (2000), and Engin and Altintas (2001)). This gives greater flexibility in the tool selection without scarifying the consistency of the model.

10.2.3 Tool Orientation and Step-over Distance Evaluation

In this work, feasible tool inclination and step-over evaluation strategies were developed. These strategies were based on effective tool profile evaluation and curvature matching methods. The following objectives were achieved in this phase;

- The developed feasible tool inclination evaluation algorithm, uses the tool model properties to ensure that the tool will always fit into the workpiece geometry, for the different tool inclination allowed during tool path generation.

- The effective tool profile evaluation strategy gives an accurate indication on the modifications required in the tool model to avoid tool placement errors.

- The developed step-over calculation algorithm ensures that the produced cusps are within the given part tolerance. This was done by linking the tool orientation and step-over to the cusp height evaluations.

10.2.4 Tool Path Generation and Gouge and Collision Evaluations

The tool path generation and evaluation phase was developed by taking the advantage of the workpiece topology as well as the data resulted from all the previous evaluation phases. A new gouge and collision detection and avoidance strategies were developed to reduce the surface errors resulting from tool placements. The following objectives were achieved in this phase;

- The developed tool placement algorithm checks the tool/workpiece contact point for placement errors using the evaluation of workpiece and tool solid model. This leads to more accurate assessment over the mathematical (e.g. Lee (1997b), Lee (1998b), Lee (1998c)) or approximated model evaluations (e.g. Oliver and Goodman (1991), Li and Jerard (1994), Jensen et al. (2002b)), in the mean time, the computational time was longer.
- The developed gouge detection strategy used the relevant part of the tool to make sure that the trailing part of the tool is not in contact with the finished workpiece surface.
- The developed gouge compensation algorithms used the feasible tool inclination data together with the gouge detection data to resolve the gouge situation by changing tool inclinations.
- The gouge detection and compensation module contributes in tool mode refinement by giving tool diameter modification values when required.
- The developed collision detection algorithm used the semifinished workpiece model together with the relevant part of tool model to check for tool collisions.
- The new collision avoidance algorithm was developed by using tool inclination range to solve the collision problem by changing the tool inclination within the feasible range.
- The collision evaluation algorithms contributes in tool model refinement process by recommending tool geometry change if it was required.

10.2.5 Tool Path Planning for Flank Milling

A solid model based flank milling module was developed to increase the robustness of this work by including other tool placements strategies. The following objectives were achieved in this phase;

- The developed curvature evaluation strategy contributed in the tool selection for this phase. This was done based on the workpiece topological evaluation to determine the tool size.
- The new algorithm developed for tool forward step evaluation makes sure that the output tool path step will produce workpiece surface within the given specifications. This reduces the surface errors due to the approximation of the tool movement by straight line segments.
- A surface deviation evaluation strategy was developed to detect the location and magnitude of maximum interference.

10.2.6 Tool Selection Process

The tool selection process was integrated with other tool path evaluation and verification processes in this work. This new approach ensures that final tool model will comply with the different constraints given to the tool model creation module from the different optimization stages of this work.

The main conclusions of this work can be summarized as follows;

1. The normal curvature evaluations gave good indication about how important a tool path evaluation position within the whole grid of positions.
2. The curvature change evaluations enhanced the workpiece interrogation process

and gave more precise indications about the importance of certain areas of the workpiece surface.

3. The developed tool model created consistent model and was successful participant in tool path generation and evaluation procedures.

4. The evaluations of feasible tool inclination range using effective tool profile algorithm gave accurate guidelines for the tool path generation methodology, avoiding the re-evaluation of tool path and tool geometry.

5. The controlled cusp height algorithm made sure that by adjusting tool path side-step the desired cusp height will be generated.

6. The developed gouge detection and avoidance algorithm eliminated the surface errors by checking for the interference of the trailing part of the tool and the finished surface of the workpiece.

7. The developed collision detection and avoidance algorithm eliminated the tool shank collision with the workpiece.

8. The developed tool model refinement process detected the potential tool path errors as a result of the tool shape. Then, the recommended changes were issued to make sure that the final tool model will produce the desired workpiece geometry.

9. The interaction and integration between the different modules of the developed system reduces the need for the iterative process of tool path generation and re-evaluations due to error resulting from tool placement and geometry.

10. Compared with commercial CAM software, the developed system was not faster in tool generation and evaluation, but the yielded workpiece topology and automatic tool selection were more precise.

10.3 Contributions

The primary contributions of this research are gaining better understanding about the tool path geometrical error sources and how to reduce these errors. Based on the presented work, the main contributions can be summarized as;

1. A solid-model based workpiece interrogation algorithm to evaluate initial iso-parametric tool path points and enhance the evaluation points locations according to the workpiece topology.
2. A robust tool model creation methodology that was able to produce consistent solid tool model with less number of parameters and can represent different milling cutter geometries.
3. A tool posture evaluation methodology using effective tool profile matching to constrain the tool orientations based on the interrogation of workpiece surface topology.
4. A step over calculation methodology based on linking the different effective tool profile with step-over evaluation to make sure that required cusp height is preserved.
5. A gouge detection and avoidance methodology, to eliminate tool path error resulting from tool back gouge.
6. A collision detection and avoidance methodology, to eliminate tool path errors due to tool shank collisions.
7. A flank milling tool path generation and checking methodology, to eliminate the forward step errors and detects tool/workpiece contact errors for developable ruled surface.
8. A multi-stage tool model refinements ensures that the final tool geometry will not be a source of tool path errors.

10.4 Recommendations for Future Work

10.4.1 Surface Configuration

This work dealt with quadri-lateral free-form surface characterized by four-sided boundaries. In industry, other classes of free-form surfaces boundaries also exist. Multi-sided surface patches has been under investigation for other applications. While this class of surface is highly complicated, its industrial applications make it an important case for extending this developed work.

10.4.2 Collision Detection and Compensation Algorithm Extension

The collision algorithm developed in this work dealt with the cutting tool model. In order for the tool path optimization algorithm to detect more potential problems, the tool holder and other parts of the machine may be integrated in this model. The addition of more components in the checking ensure that the calculated tool path will anticipate more problems before generating the final tool path.

10.4.3 Multi-Tool Selection

The developed work dealt with the tool selection as an integrated part of the tool path generation and verification system. This improves the cutting efficiency and reduces the tool path error. This work can be extended by considering multiple tool changes in tool path generation process. This can be achieved by dividing the workpiece surface into sub patches. The sub patches can be treated separately which permit the use of larger tool size in the areas that do not require small tool. This approach minimizes the machining time and increase the metal removal rate.

Some problems must be addressed before implementing this algorithm. The transition areas between the surface patches will have high cusp values, and surface errors. The tool movement control algorithm must be developed to ensure that the tool exit path at the surface boundaries will preserve the neighboring surface patch shape.

10.4.4 Flank Milling of Non-developable Surfaces

The developed flank milling planning module checks and compensate for tool path errors resulting from tool advancement scheme. It also detects the errors along tool/workpiece contact line. This work dealt with developable rule surface.

This work can be extended by the development of error compensation algorithm to reduce the surface error. The proposed algorithm uses the principal curvature at both contact points at the ends of the surface. The direction of the maximum curvature indicates the direction of the correction. By using the error data obtained from the error evaluation module together with the curvature direction vectors, the tool orientation can be altered to place the tool where maximum deviation was detected. This strategy helps reduce the interference between the tool and workpiece and permits for further finishing operation based on the re-evaluation of the error after the tool re-orientation. This strategy should give similar results as the research by Menzel et al. (2004), shown in figure 2.21(b), but using exact tool/workpiece models.

10.4.5 Integration of Five Axis Machining Process Evaluation

While this work dealt with the optimization of the geometric aspects of tool path planning process, other aspects must be considered.

Several published researches optimize the multi axis milling process. In the work by Veldhuis (1998), the multi axis machining errors due do thermal expansion were investigated. The compensation algorithms to reduce thermal effects were developed. In the work by Abrari (1998), the effect of the deflection on thin workpiece was investigated. The compensation algorithms to reduce workpiece errors were also developed. In the work by Bailey (2001)), the optimization algorithms for the cutting forces during fives axis milling were developed. In the work by Tounsi and Elbestawi (2003a), and Tounsi and Elbestawi (2003b) , the tool path optimization included the elimination of the machine tool feed system errors by altering the tool path segments and feed scheduling.

These research works had the same goal as the work developed in this thesis, optimization of five axis milling process, but from different aspects. The output from this work can be used as an input to these different optimization stages to reach more global optimization state of the process.

Other research-in-progress include the inspection planning of free-form surface, and the error characterization for the free-form surface, uses the same development environment of the current work and can integrate the inspection process in the optimization of five axis machining.

BIBLIOGRAPHY

Abdullahil, A., Feng, H. Y., and Wang, L. (2004). Simplified and efficient calibration of a mechanistic cutting force model for ball-end milling. *International Journal of Machine Tools and Manufacture*, 44(2):291–298.

Abrari, F. (1998). *Multi-Axis Milling of Flexible Parts.* PhD thesis, McMaster University, Hamilton, Ontario, Canada.

Abrari, F., Elbestawi, M. A., and Spence, A. D. (1998). On the dynamics of ball end milling - modelling of cutting forces and stability analysis. *International Journal of Machine Tools and Manufacture*, 38(3):215–237.

Anderson, R. O. (1987). Detecting and eliminating collisions in NC machining. *Computer-Aided Design*, 10(4):231–237.

Armerago, E. J. A. and Deshpande, N. P. (1989). Computerized predictive cutting models or forces in end milling including eccentricity effects. *Annals Of The CIRP*, 38:45–49.

Austin, S. P., Jerard, B., R., and Drysdale, R. L. (1996). Comparison of discretization algorithms for nurbs surfaces with application to numerically controlled machining. *Computer-Aided Design*, 28:71–83.

Bailey, T. E. (2001). *Generic Mechanistic Modeling For Multi-Axis Machining.* PhD thesis, McMaster University, Hamilton, Ontario, Canada.

Balasubramaniam, M., Ho, S., Sarama, S., and Adachi, Y. (2002). Generation of collision−free 5−axis tool paths using haptic surface. *Computer-Aided Design*, 34(4):267–279.

Barnhill, R. E., Farin, G., and Hogen, H. (1988). Twists, curvatures and surface interrogation. *Computer-Aided Design*, 20(6):341–346.

Bedi, S., Mann, S., and Menzel, C. (2003). Flank milling with flat end milling cutter. *Computer-Aided Design*, 35(3):293–300.

Bohez, E. J., Senadhera, S. D., Pole, K., Duflou, J. R., and Tar, T. (1997). A geomtric modeling and five−axis machining algorithm for centrifugal impellers. *Journal of Manufacturing Systems*, 16(6):422–436.

Böhm, W., Farin, G., and Kahmann, J. (1984). A survey of curve and surface methods in CAGD. *Computer-Aided Geometric Design*, 1:1–60.

Budak, E. (2000). Improving productivity and part quality in milling of titanium based impellers by chatter suppression and force control. *Annals of the CIRP*, 49(1):31–36.

Capello, E. and Semeraro, Q. (2001). The harmonic fitting method for assessment of the substitute geometry estimate error. part ii: statistical approach, machining process analysis and inspection plan optimization. *International Journal of Machine Tools and Manufacture*, 41:1103–1129.

Chappel, I. T. (1983). Use of vectors to simulate material removed by NC milling. *Computer-Aided Design*, 15(3):156–158.

Chen, Y. J. and Ravani, B. (1987). Offset surface generation and contouring in computer-aided design. *ASME Journal of Mechanisms, Transmissions and Automation Design*, 109:132–142.

Cho, H. D., Jun, Y. T., and Yang, M. Y. (1993). Five-axis milling for effective machining of sculptured surfaces. *International Journal of Production Research*, 31(11):2559–2573.

Choi, B. K. and Jun, C. S. (1989). Ball-end cutter interference avoidancein NC machining of sculptured surfaces. *Computer-Aided Design*, 21(6):371–378.

Clayton, P. A., El Wardany, T., Elbestawi, M. A., and Viens (2000). A mechanistic force model of five axis milling process. *Proceedings of the ASME, Med*, 11:979–987.

Deng, Z., Leu, M. C., L., W., and Blackmore, D. (1996). Determination of flat-end cutter orientation in 5-axis machining. *ASME Journal of Manufacturing Science and Engineering*, 4:73–80.

Drysdale, R. L., Jerard, R. B., Schaudt, B., and Hauck, K. (1989). Discrete simulation of NC machining. *Algorithmica*, 4:33–60.

El Mounayri, H., Spence, A. D., and Elbestawi, M. A. (1998). Milling process simulation - a generic solid modeller based paradigm. *ASME Journal of Manufacturing Science and Engineering*, 120(2):213–221.

Elber, G. and Cohen, E. (1994). Tool path generation for freeform surface models. *Computer-Aided Design*, 26(6):490–496.

ElKott, D., ElMaraghy, H., and ElMaraghy, W. (2002). Automatic sampling for CMM inspection planning of free-form surfaces. *International Journal of Production Research*, 40(11):2653–2676.

Engin, S. and Altintas, Y. (2001). Mechanics and dynamics of general milling cutters part i: helical end mills. *International Journal of Machine Tools and Manufacture*, 41:2195–2212.

Farin, G. (1993). *Curves and surfaces for computer aided geometric design: A Practical giude*. Academic press, Boston.

Feng, H. and Li, h. (2000). NC tool planning for sculptured surface machining based on constant scallop height. In *Transaction of NAMRI*, volume 28, pages 377–382, Lexington, KY.

Fussell, B., Jerard, R., and Hemmett, J. G. (2003). Modeling of cutting geometry and forces for 5-axis sculptured surface machining. *Computer-Aided Design*, 35(4):333–346.

Gradisek, J., Kalveram, M., and Weinert, K. (2004). Mechanistic identification of specific force coefficients for a general end mill. *International Journal of Machine Tools and Manufacture*, 44(4):401–414.

Gu, F., Melkote, S. N., Kapoor, G., and Devor, R. (1997). A model for the prediction of surface flatness in face milling. *Journal of Manufacturing Science and Engineering*, 119(4):476–484.

Ho, M. C., Hwang, Y. R., and Hu, C. H. (2003). Five−axis tool orientation smoothing using quaternion interpolation algorithm. *International Journal of Machine Tools and Manufacture*, 43:1259–1267.

Hoffmann, C. (1989). *Geometric and solid modeling - An introduction*. Morgan Kaufmann publishers.

Huang, Y. and Ovliver, J. (1992). Non-constant parameter NC tool path generation on sculptured surfaces. *International Journal of Advanced Manufacturing Technology*, 9:281–290.

Imani, B. (1998). *Model Based Die Cavity Machining Simulation Methodology*. PhD thesis, McMaster University, Hamilton, Ontario, Canada.

Jensen, C. G. and Anderson, D. C. (1992). Accurate tool placement and orientation for finish surface machining. *Concurrent Engineering, ASME PED*, 59:127– 145.

Jensen, C. G., Mulkay, E. L., and Simpson, T. S. (1995). A practical implementation of curvature matched machining for complex geometric objects. *Concurrent Product and Process Engineering*, 85:1–14.

Jensen, C. G., Red, W. E., and Pi, E. (2002a). Tool selection for five−axis curvature matched machining. *Computer-Aided Design*, 34:251–266.

Jensen, C. G., Red, W. E., and Pi, J. (2002b). selection for five-axis curvature matched machining. *Computer-Aided Design*, 34(3):251–266.

Jun, C., Cha, K., and Lee, Y. S. (2003). Optimizing tool orientations for 5-axis machining by configuration-space (c-space) search method. *Computer-Aided Design*, 35(6):549–566.

Karunakaran, K. P., Shanmuganathan, P. V., Gupta, N., and Issac, M. (2000). Swept volume of a generic cutter. *Journal of Engineering Manufacture*, 214(915-938).

Kim, B. and Chu, C. (1994). Effect of cutter mark on surface roughness and scallop height in sculptured surface machining. *Computer-Aided Design*, 26(3):179–188.

Kim, K. I., Chon, J., Y., and Kim, K. (1996). Simulation and verification of CNC tool path for sculptured surfaces. In *Transaction of NAMRI*, volume 24, pages 69–74, Minneapolis, MN.

Kim, K. I. and Kim, K. K. (1995). A new machine strategy for sculptured surfaces using offset surface. *International Journal of Production Research*, 33:1683–1697.

Larue, A. and Anselmetti, B. (2003). Deviation of a machined surface in flank milling. *International Journal of Machine Tools and Manufacture*, 43:129–138.

Lee, K. (1999). *Principles of CAD/CAM/CAE systems.* Addison Wesley Longman.

Lee, P. and Altintas, Y. (1996). Prediction of ball−end milling forces from orthogonal cutting data. *International Journal of Machine Tools and Manufacture*, 36:1059–1072.

Lee, Y. S. (1997a). Admissible tool orientation control of gouging avoidance for 5-axis complex surface machining. *Computer-Aided Design*, 29(7):507–521.

Lee, Y. S. (1997b). Filleted end mill placement problems and error analysis for multi-axis CNC machining. In *Transaction of NAMRI*, volume 25, Lincoln, NE.

Lee, Y. S. (1998a). Adaptive tool path planning by machining strip evaluation for 5-axis complex surface machining. In *Transactions of NAMRI*, volume 26, pages 141–146, Atlanta, GA.

Lee, Y. S. (1998b). Mathematical modeling of using different endmills and tool placement problems for 4- and 5-axis NC complex surface machining. *International Journal of Production Research*, 36(3):785–814.

Lee, Y. S. (1998c). Non- isoparametric tool path planning by machining strip evaluation for 5-axis sculptured surface machining. *Computer-Aided Design*, 30(7):559–570.

Lee, Y. S. and Chang, T. C. (1996). Machined surface error analysis for 5-axis machining. *International Journal of Production Research*, 34(1):111–135.

Lee, Y. S. and Chang, T. C. (1997). Admissible tool orientation control of gouging avoidance for 5-axis complex surface machining. *Computer-Aided Design*, 29(7):507–527.

Leu, M., Wang, L., and Blackmore, D. (1997). A verification program for 5-axis NC machining with general apt tools. *Annals of the CIRP*, 46(1):419–424.

Li, S. X. and Jerard, R. B. (1994). 5-axis machining of sculptured surfaces with a flat-end cutter. *Computer-Aided Design*, 26(3):165–178.

Lim, M. E., Feng, H., and Menq, C. (1993). The prediction of dimensional errors for machining sculptured surfaces using ball-end milling. *Manufacturing Science and Engineering, ASME*, 64:149–156.

Liu, X. W. (1995). Five–axis NC cylindrical milling of sculptured surfaces. *Computer-Aided Design*, 27(12):887–894.

Mäntylä, M. (1988). *An introduction to solid modeling.* Computer science press.

Marciniak, K. (1991). *Geometric Modelling for Numerically Controlled Machining.* Oxford university press.

Marshall, S. and Griffiths, J. (1994). A survey of cutter path construction techniques for milling machines. *International Journal of Production research*, 32(12):2861–2877.

Melkote, S. N. and Kendall, L. A. (1989). End-mill cutter engagement determination from CNC tool trajectories for planar surface generation. In *Transaction of NAMRI*, volume 17, Dearborn, Mi.

Menzel, C., Bedi, S., and Mann, S. (2004). Triple tangent flank milling of ruled surfaces. *Computer-Aided Design*, 36(3):289–296.

Monies, F., Felices, J. N., Rubio, W., and Redonnet, J.-M. (2002). Five−axis NC milling of ruled surfaces:optimal geometry of a conical tool. *International Journal of production research*, 12:2901–2922.

Monies, F., Redonnet, J.-M., Rubio, W., and Lagarrigue, P. (2000). Improved positioning of conical mill for machining ruled surfaces: application to turbine blades. *Journal of Engineering Manufacture*, 214(625-634).

Morishige, K., Takeuchi, Y., and Kase, K. (1997). Tool path generation using c-space for 5-axis machining. *Manufacturing Science and Technology*, 6(2):333–340.

Oliver, J. H. and Goodman, E. D. (1991). Direct dimensional NC verification. *Computer-Aided Design*, 22(1):3–9.

Patrikalakis, N. and Maekawa, T. (2000). *Shape interrogation for computer aided design and manufacturing.* Springer-Verlag.

Pi, J., Red, E., and Jensen, C. G. (1998). Grind-free tool path generation for five-axis surface machining. *Computer Integrated Manufacturing*, 11(4):337–350.

Piegl, L. (1991). On nurbs: A survey. *IEEE Computer Graphics and applications*, 11:55–71.

Piegl, L. and Tiller, W. (1997). *The NURBS Book*. Springer-Verlag.

Ramesh, R., Mannan, M., and Poo, A. (2000). Error compensation in machine tools-a review. part i: Geometric , cutting-force included and fixture-dependent errors. *International Journal of Machine Tools and Manufacture*, 40:1235–1256.

Rao, N., Ismail, F., and Bedi, S. (2000). Integrated tool positioning and tool path planning for five-axis machining of sculptured surfaces. *International journal of production research*, 38(12):2709–2724.

Redonnet, J. M., Rubio, W., and Dessein, G. (1998). Side milling of ruled surfaces: optimum positioning of milling cutter and calculation of interference. *International Journal of Advanced Manufacturing Technology*, 14:459–465.

Rehsteiner, F. and Renker, H. J. (1993). Collision free five-axis milling of twisted ruled surfaces. *Annals of the CIRP*, 42(1):457–460.

Roth, D., Ismail, F., and Bedi, S. (2003). Mechanistic modelling of the milling process using an adaptive depth buffer. *Computer-Aided Design*, 35(14):1287–1303.

She, C. H. and Lee, R. S. (2000). A postprocessor based on the kinematics model for general five axis machine tool. *SME, Journal of manufacturing processes*, 2(2):131–138.

Sim, C. and Yang, M. (1993). the prediction of the cutting force in ball−end milling with flexible cutter. *International Journal of Machine Tools and Manufacture*, 33:267–284.

Spence, A. D. and Altintas, Y. (1994). A solid modeler based milling process simulation and planning system. *ASME Journal of Engineering for Industry*, 116:61–69.

Stute, G. and Storr, A., a. S. W. (1979). NC programming of ruled surfaces for five axis machining. *Annals of the CIRP*, 28(1):267–271.

Tounsi, N. and Elbestawi, M. A. (2003a). Optimized feed scheduling in three axes machining. part i: Fundamentals of the optimized feed scheduling strategy. *International Journal of Machine Tools and Manufacture*, 43(3):253–267.

Tounsi, N. and Elbestawi, M. A. (2003b). Optimized feed scheduling in three axes machining part ii: Experimental validation. *International Journal of Machine Tools and Manufacture*, 43(3):269–282.

Tounsi, N., ElKott, D., and Elbestawi, M. (2002). Inspection of sculptured surfaces for closed loop machining. In *The 3rd CIRP International Seminar on Intelligent Computation in Manufacturing Engineering (ICME 2002)*, Naples, Italy.

Varady, T. and Pratt, M. (1984). Design techniques for the definition of solid objects with free form geometry. *Computer-Aided Geometric Design*, 1:207–225.

Veldhuis, S. C. (1998). *Modelling and compensation of errors in five-axis machining.* PhD thesis, McMaster University, Hamilton, Ontario, Canada.

Vickers, G., Ly, M., and Oetter, R. (1990). *Numerically Controlled Machine Tools.* Ellis Harwood, New York.

Vikers, G. and Quan, K. (1989). Ball end mills versus end mills for curved surface machining. *Journal of engineering for industry*, 111:22–26.

Voelcker, H. B. and Hunt, W. A. (1981). The role of solid modelling in machining-process modelling and nc verification. In *SAE International Congress and Exposition.*

Wang, w. P. (1988). Modelling for optimizing metal removal of three dimensional end-milling. *Journal of manufacturing systems*, 7:57–65.

Wang, w. P. and Wang, K. K. (1986). Geometric modeling for swept volume of moving solids. *IEEE Computer Graphics and Applications*, 12:8–17.

Yamazaki, K., Kojma, C., Sakamoto, C., and Saito, T. (1991). Real–time model reference adaptive control of 3d sculptured surface machining. *Annals Of The CIRP*, 40:479–482.

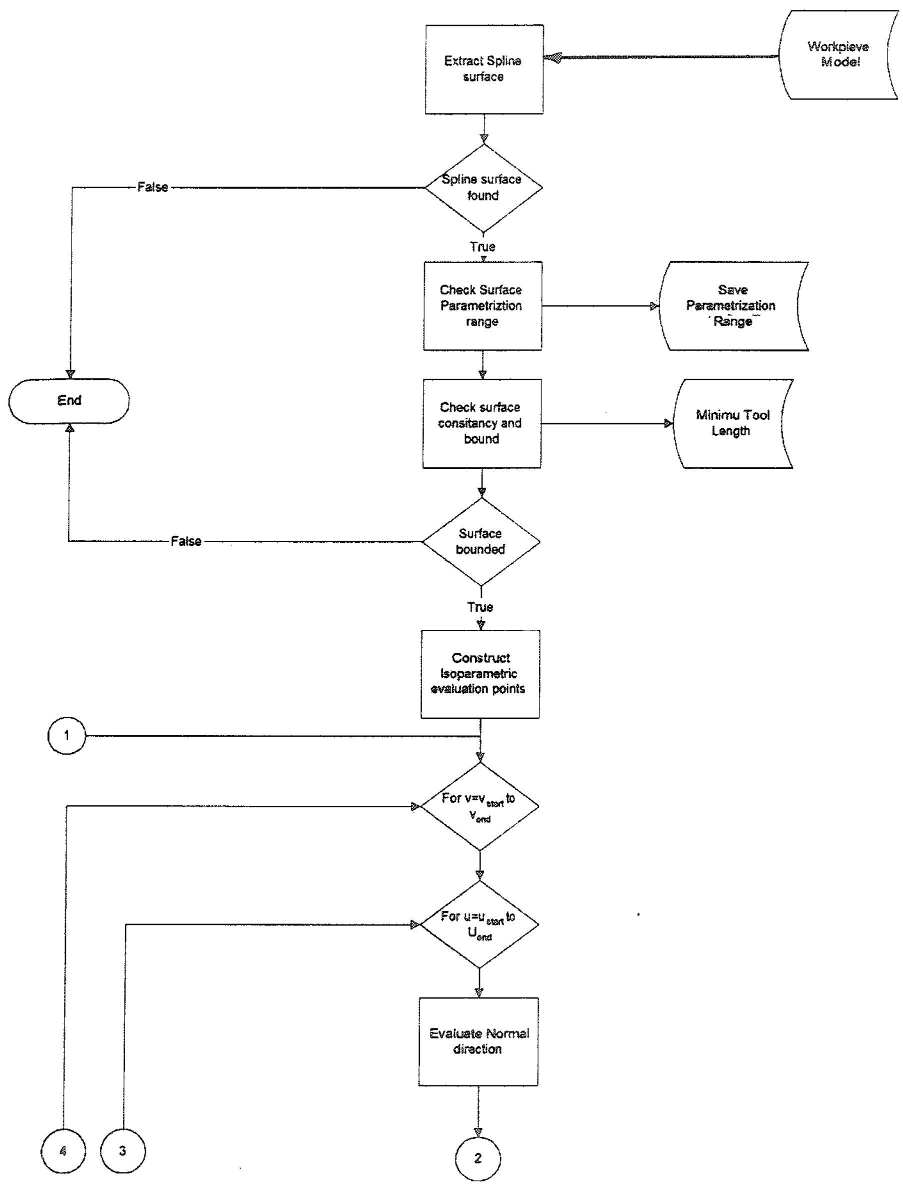
Zeid, I. (1991). *CAD/CAM heory and practice.* McGraw-Hill.

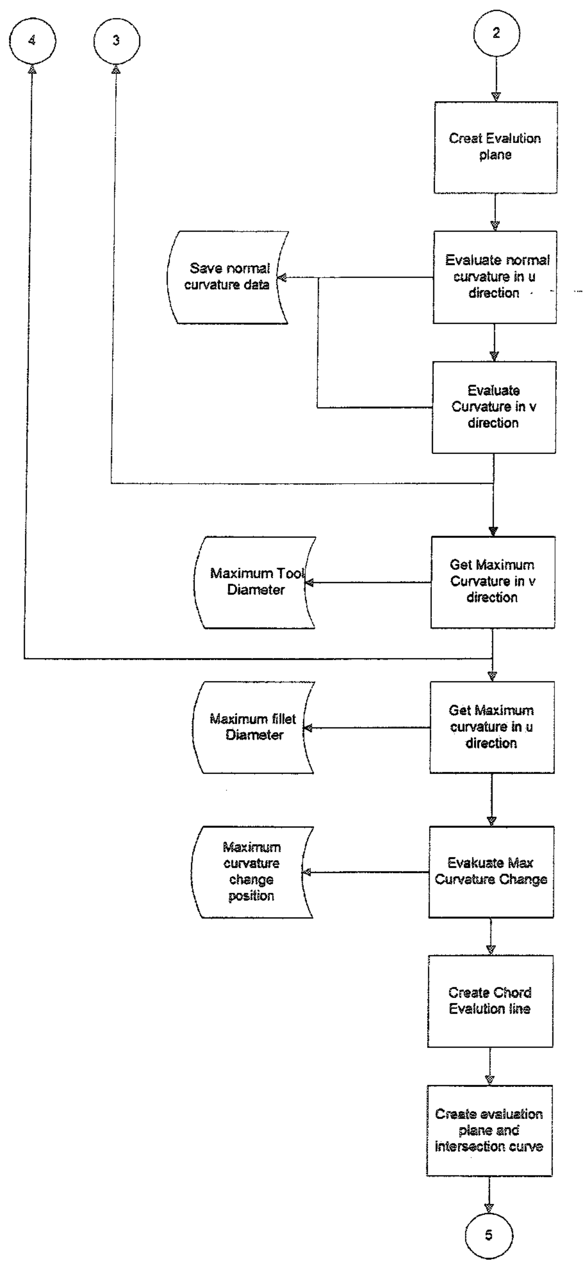
Zhou, L. and Lin, Y. J. (2001). An effective globle gouge detection in tool path planning for freeform surface machining. *International Journal of Advanced Manufacturing Technology*, 18:461–473.

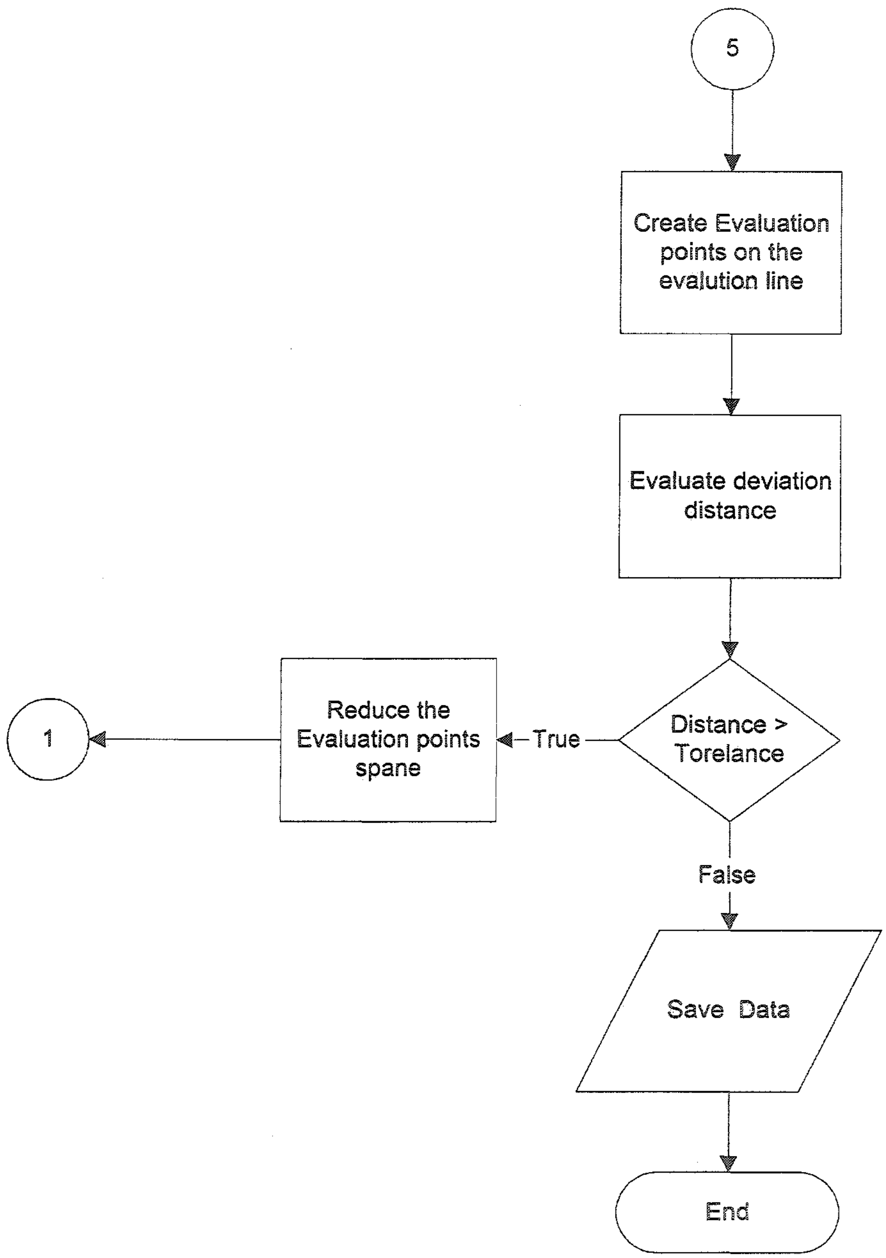
APPENDIX A

Flow Charts for the developed algorithms

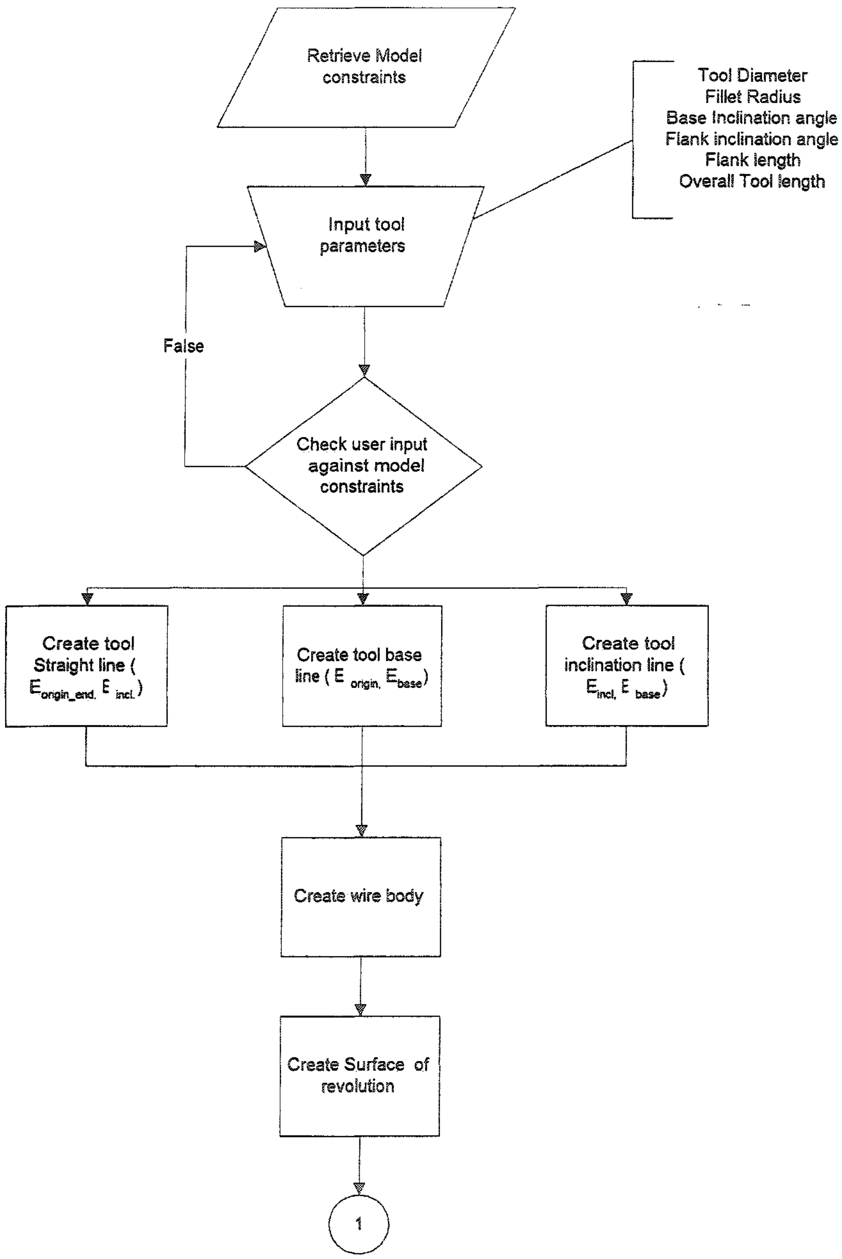
A.1 Curvature Evaluation

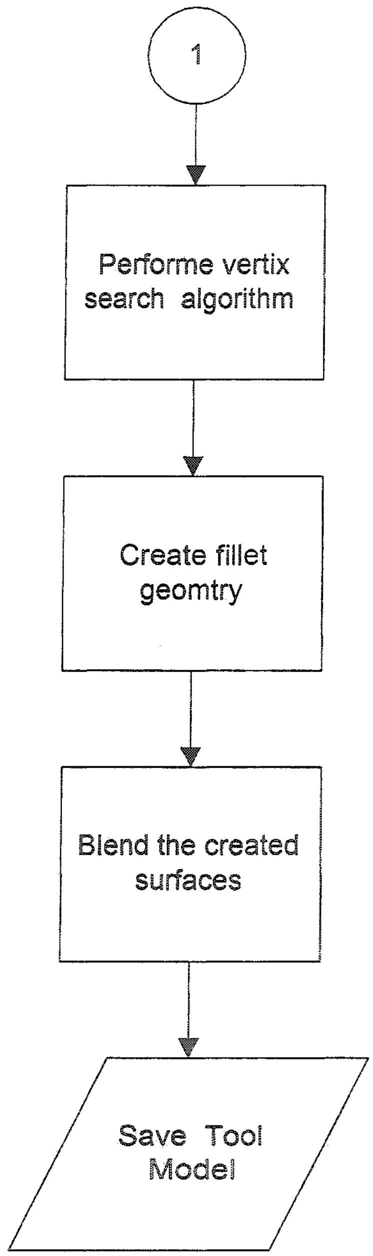



4
3
2
Creat Evalution plane
Evaluate normal curvature in u direction
Save normal curvature data
Evaluate Curvature in v direction
Get Maximum Curvature in v direction
Maximum Tool Diameter
Get Maximum curvature in u direction
Maximum fillet Diameter
Evakuate Max Curvature Change
Maximum curvature change position
Create Chord Evalution line
Create evaluation plane and intersection curve
5

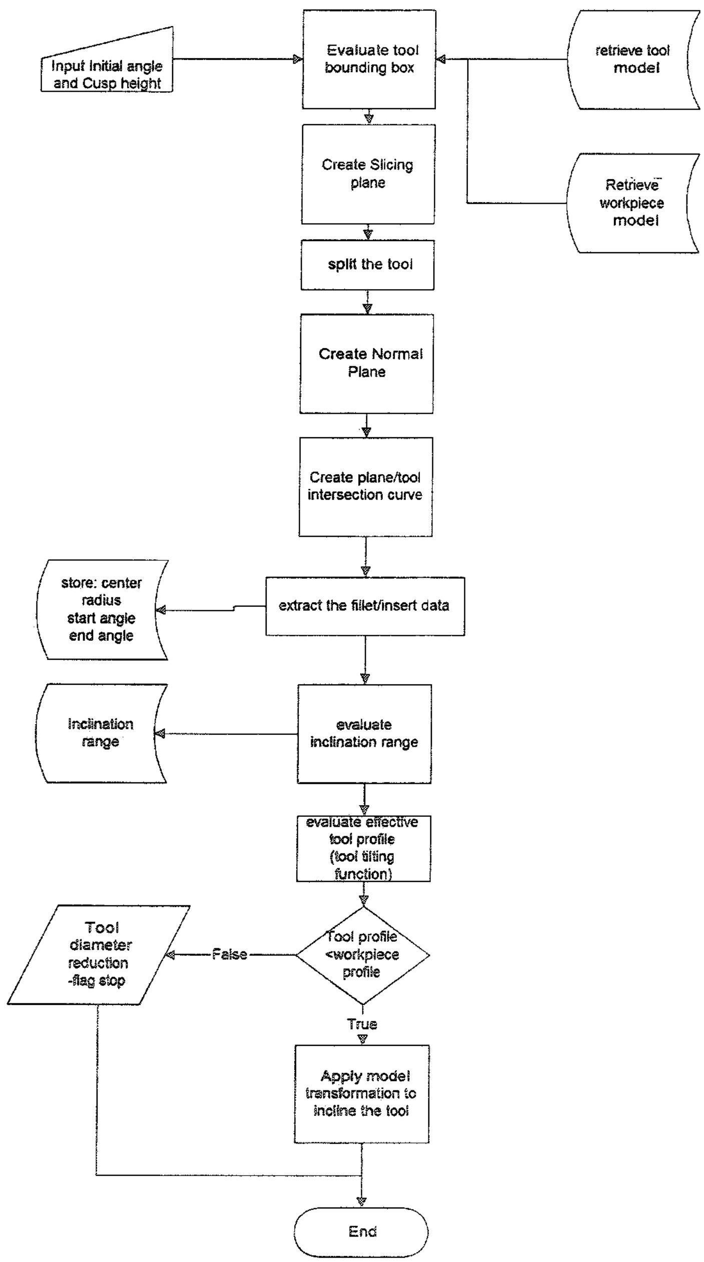

5
Create Evaluation points on the evalution line
Evaluate deviation distance
Distance > Torelance
True
Reduce the Evaluation points spane
1
False
Save Data
End

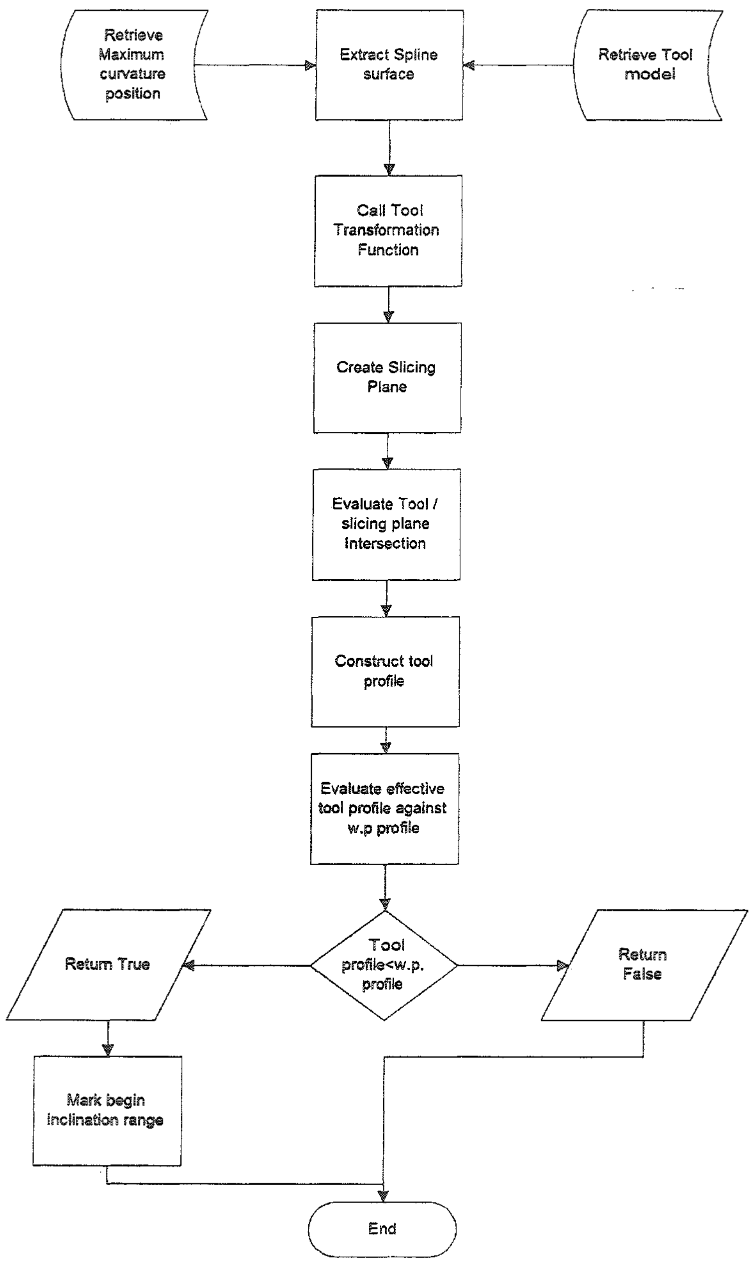
A.2 Tool Model

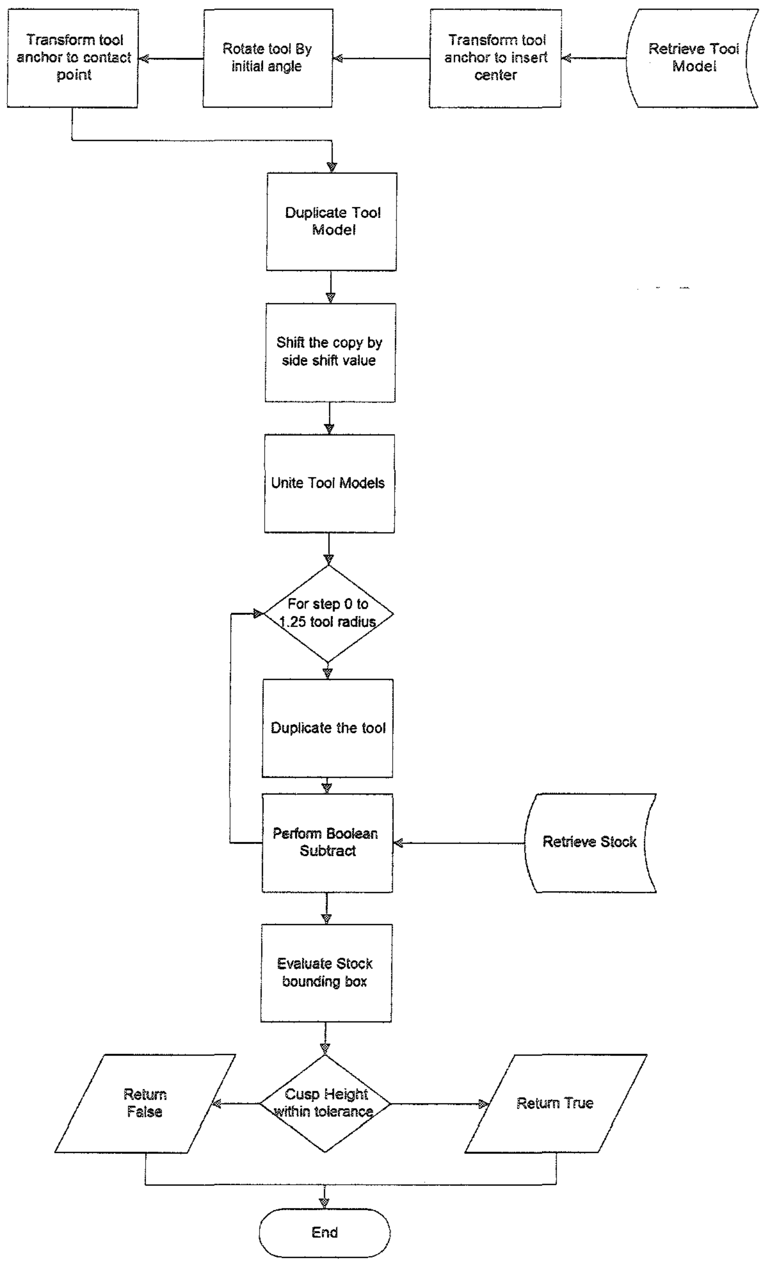


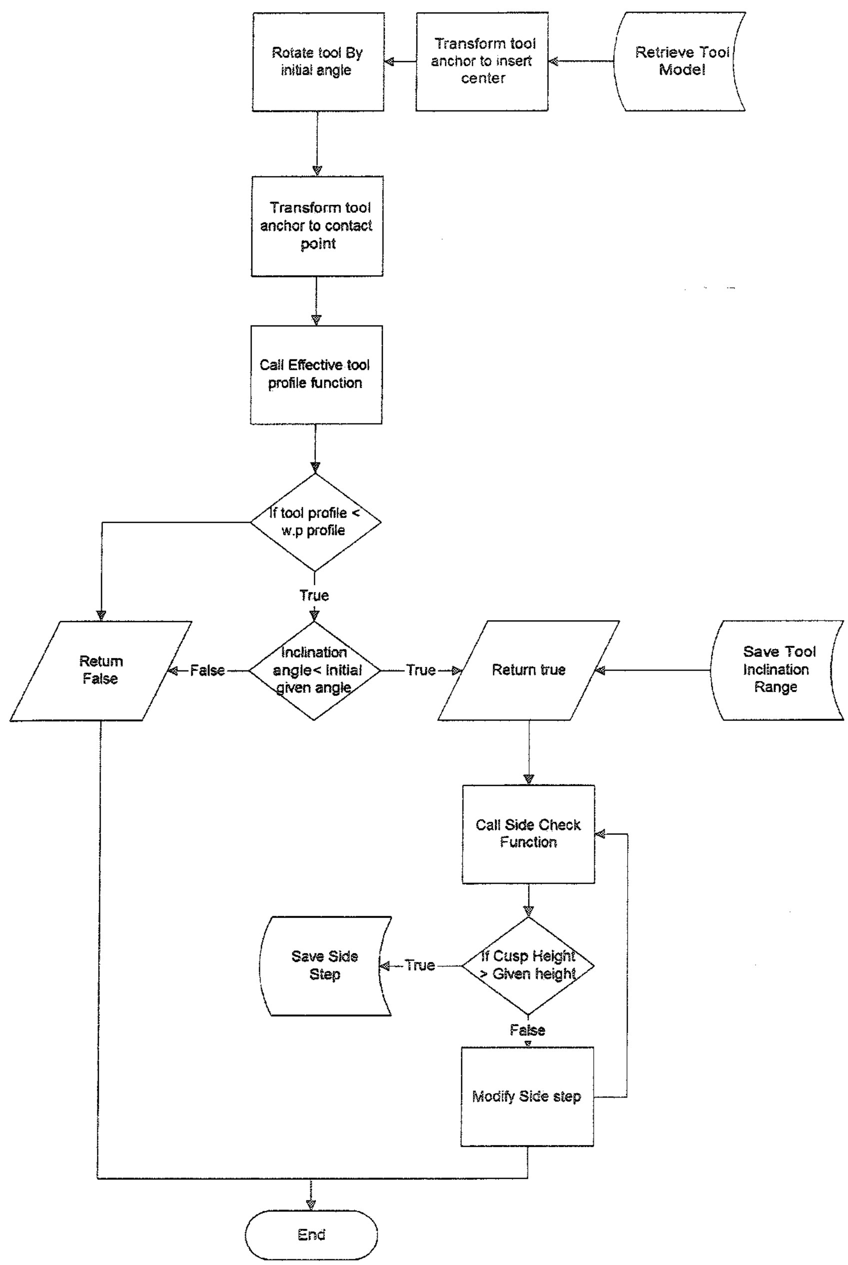

1
Performe vertix search algorithm
Create fillet geomtry
Blend the created surfaces
Save Tool Model

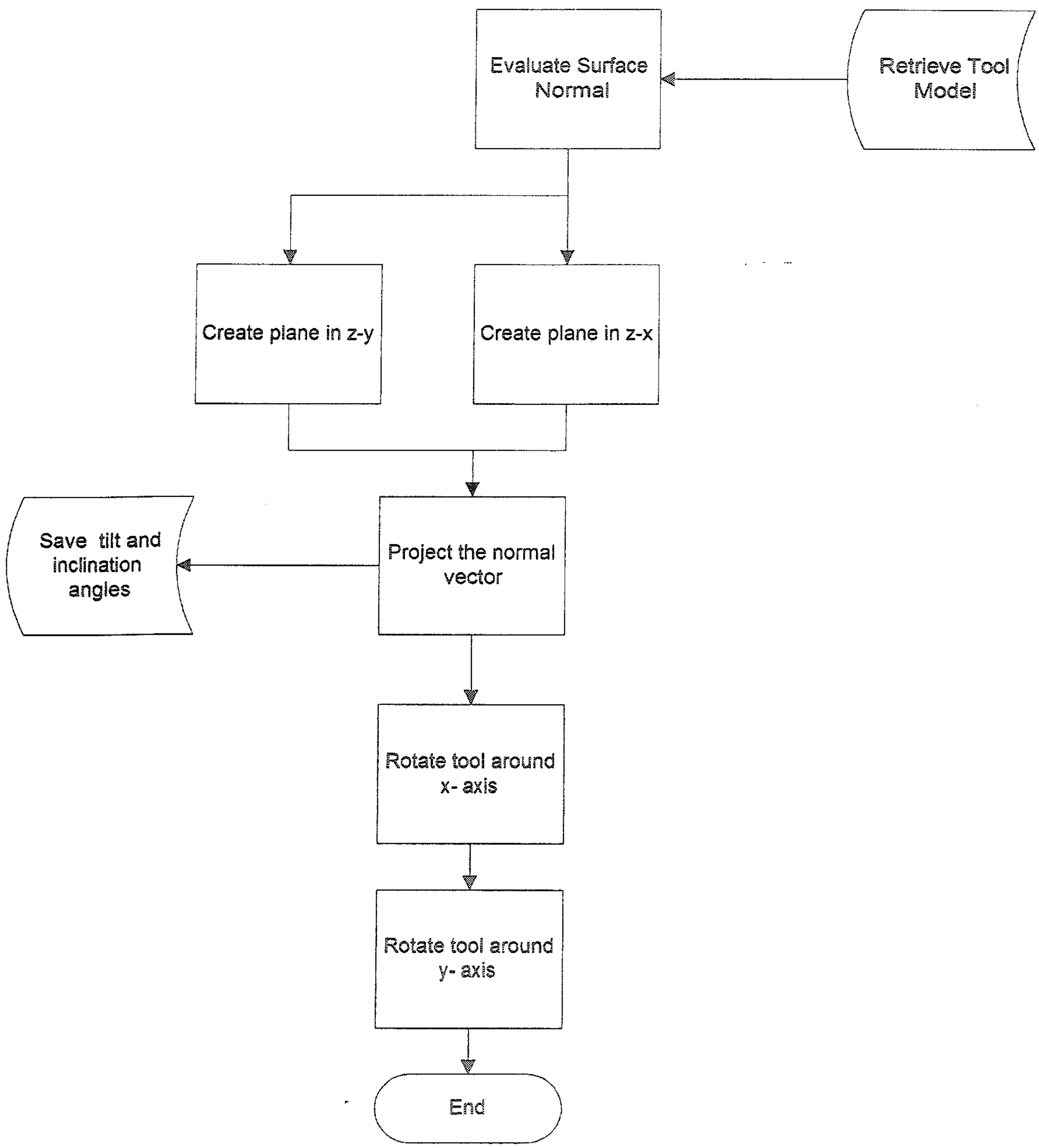
A.3 Tool Positioning



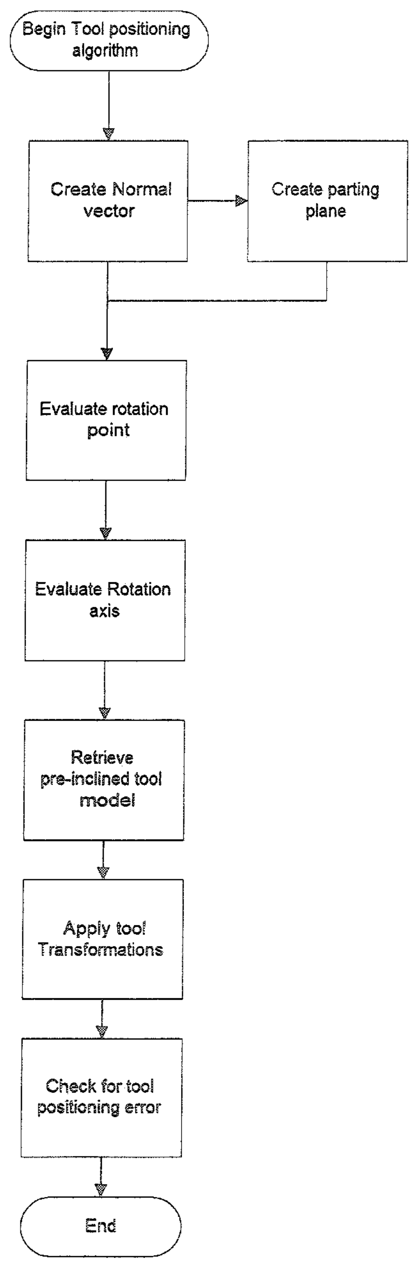

Retrieve Maximum curvature position
Extract Spline surface
Retrieve Tool model
Call Tool Transformation Function
Create Slicing Plane
Evaluate Tool / slicing plane Intersection
Construct tool profile
Evaluate effective tool profile against w.p profile
Tool profile<w.p. profile
Return True
Return False
Mark begin inclination range
End

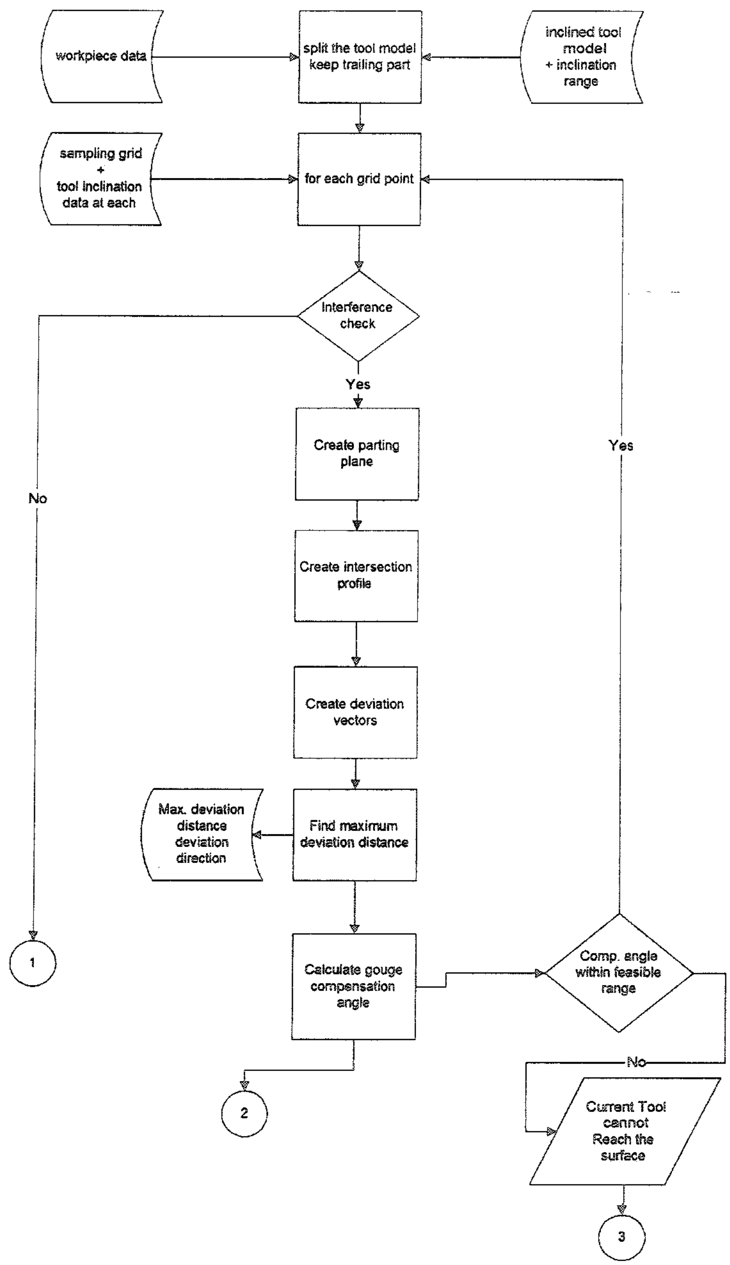

Transform tool anchor to contact point
Rotate tool By initial angle
Transform tool anchor to insert center
Retrieve Tool Model
Duplicate Tool Model
Shift the copy by side shift value
Unite Tool Models
For step 0 to 1.25 tool radius
Duplicate the tool
Perform Boolean Subtract
Retrieve Stock
Evaluate Stock bounding box
Return False
Cusp Height within tolerance
Return True
End

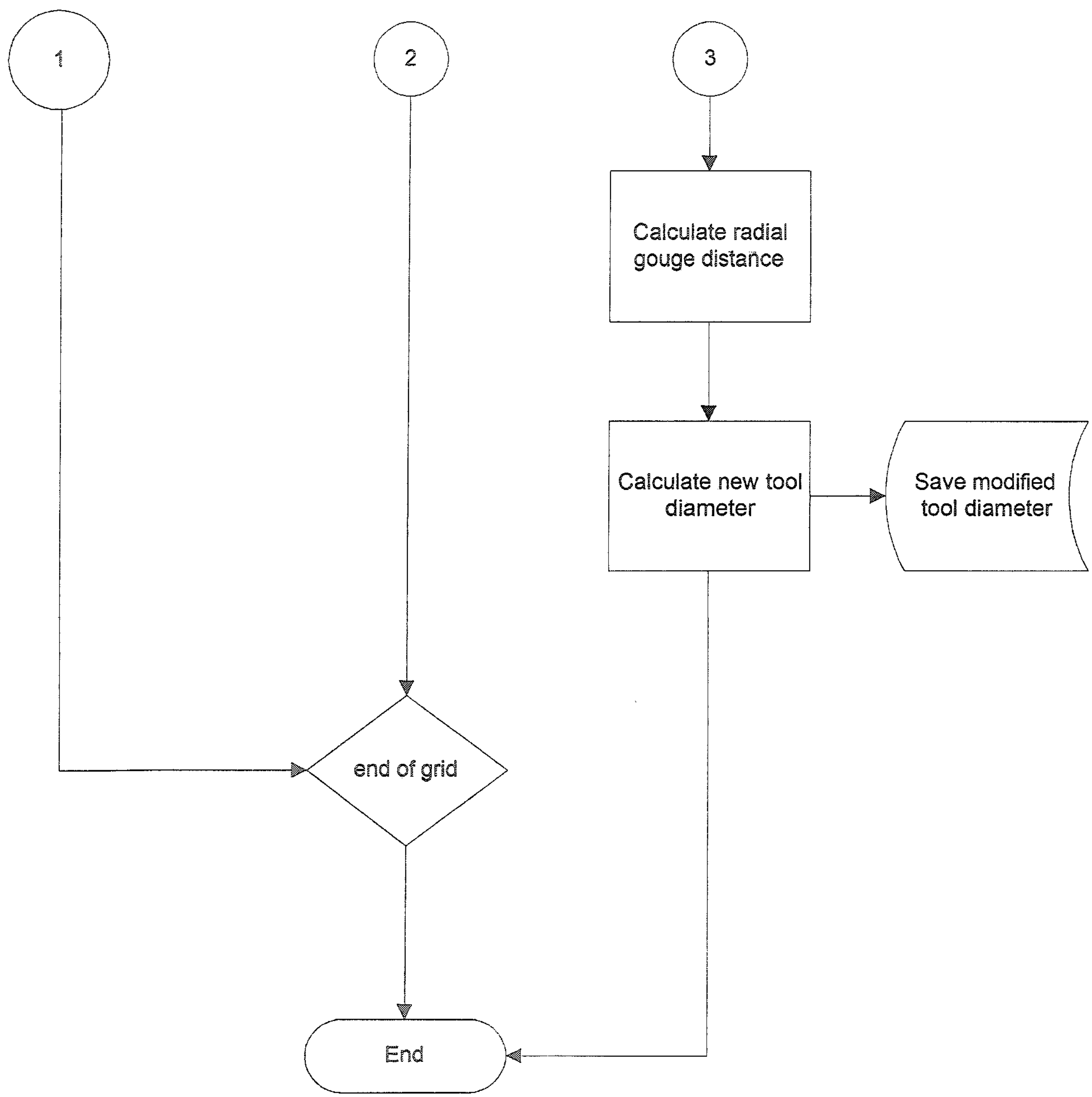

Rotate tool By initial angle
Transform tool anchor to insert center
Retrieve Tool Model
Transform tool anchor to contact point
Call Effective tool profile function
If tool profile < w.p profile
True
Return False
False
Inclination angle< Initial given angle
True
Return true
Save Tool Inclination Range
Call Side Check Function
Save Side Step
True
If Cusp Height > Given height
False
Modify Side step
End

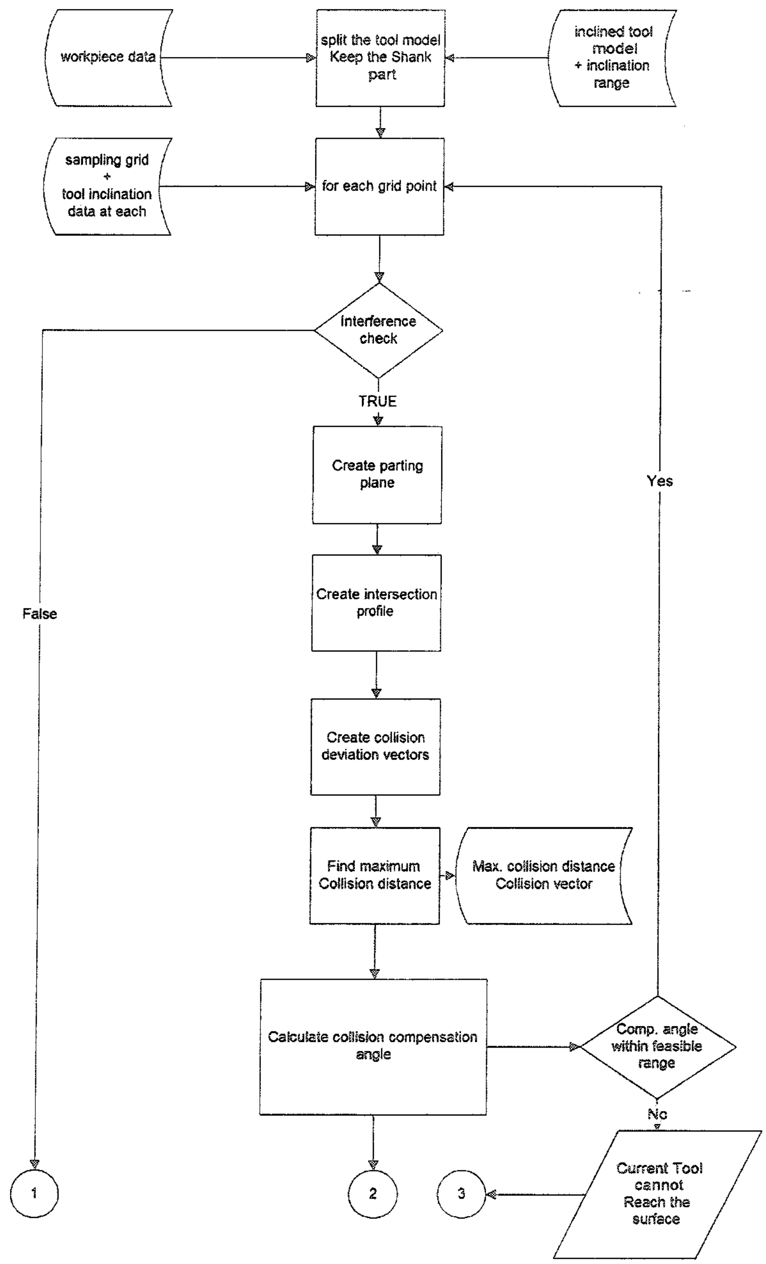

Evaluate Surface Normal
Retrieve Tool Model
Create plane in z-y
Create plane in z-x
Save tilt and inclination angles
Project the normal vector
Rotate tool around x- axis
Rotate tool around y- axis
End

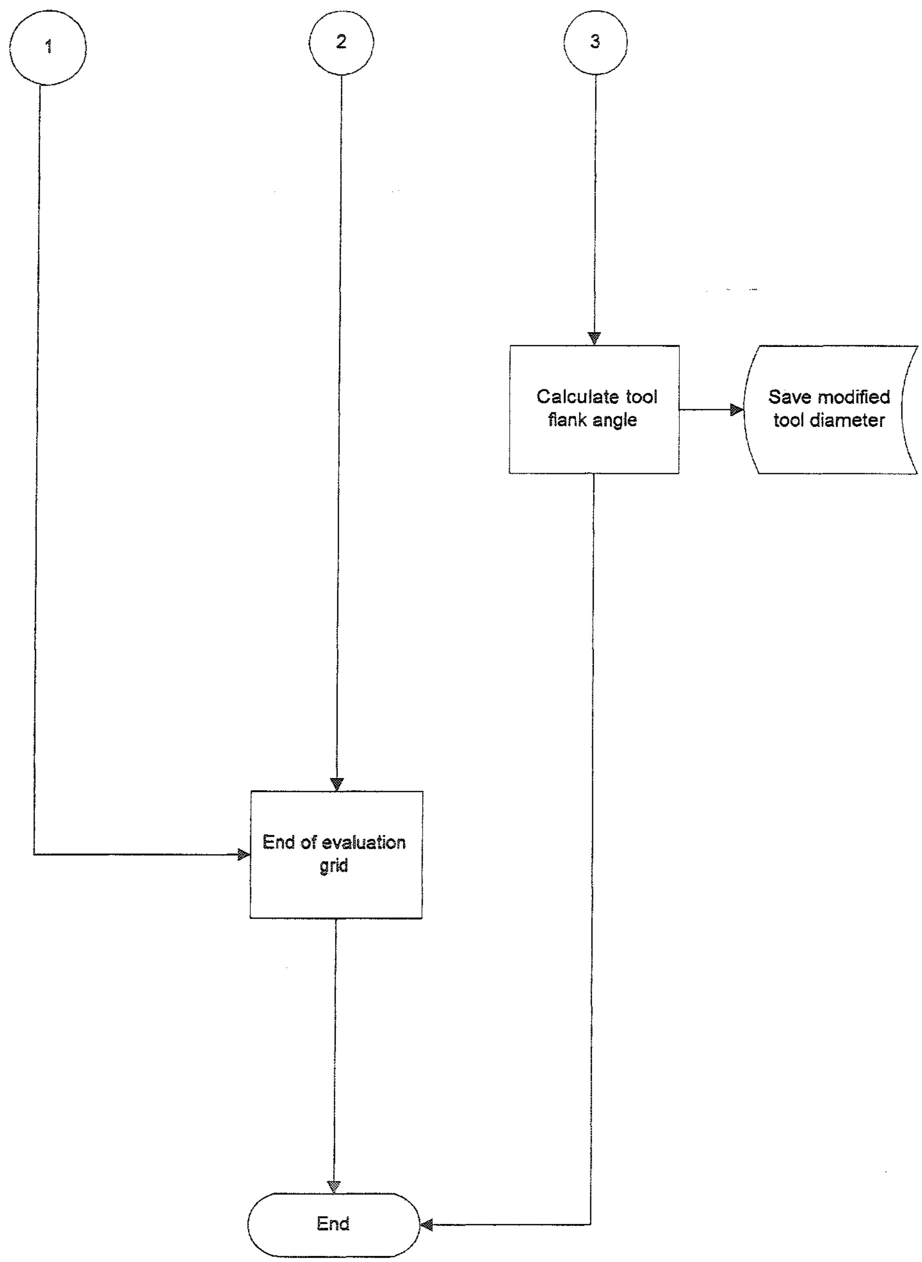
A.4 Tool Path Generation and Gouge and Collision Evaluations



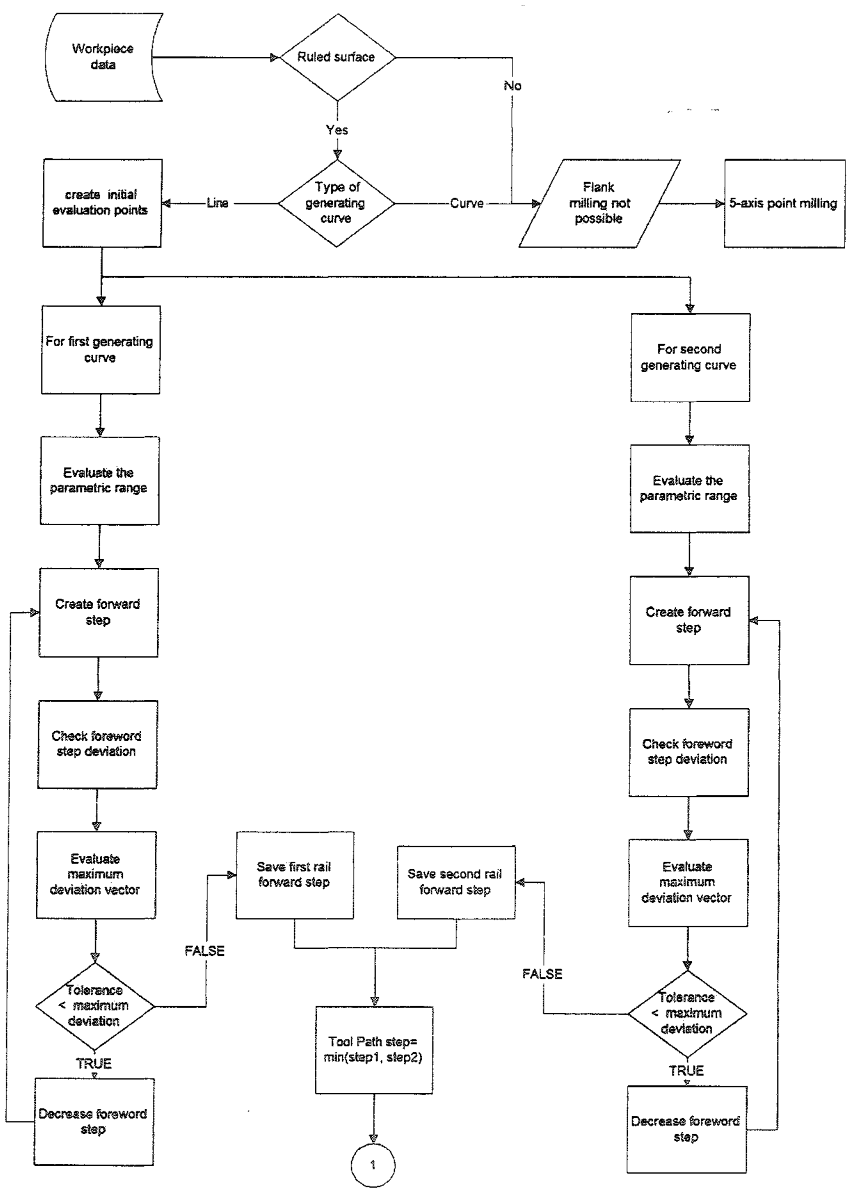

workpiece data
split the tool model keep trailing part
inclined tool model + inclination range
sampling grid + tool inclination data at each
for each grid point
Interference check
Yes
No
Create parting plane
Create intersection profile
Create deviation vectors
Max. deviation distance deviation direction
Find maximum deviation distance
Calculate gouge compensation angle
Comp. angle within feasible range
Yes
No
Current Tool cannot Reach the surface
1
2
3

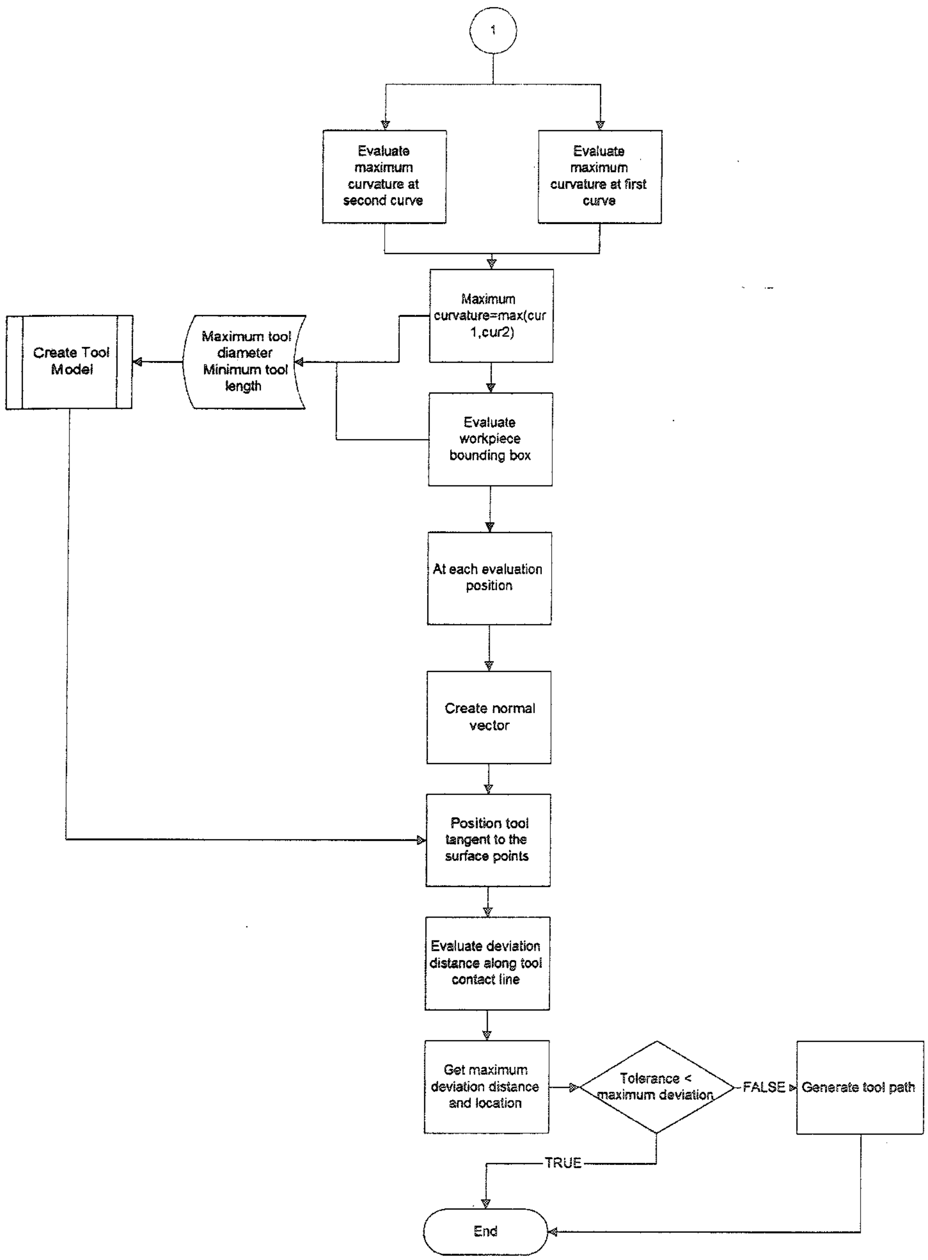

1
2
3
Calculate radial gouge distance
Calculate new tool diameter
Save modified tool diameter
end of grid
End


workpiece data
split the tool model Keep the Shank part
inclined tool model + inclination range
sampling grid + tool inclination data at each
for each grid point
Interference check
TRUE
False
Yes
Create parting plane
Create intersection profile
Create collision deviation vectors
Find maximum Collision distance
Max. collision distance Collision vector
Calculate collision compensation angle
Comp. angle within feasible range
No
Current Tool cannot Reach the surface
1
2
3


1
2
3
Calculate tool flank angle
Save modified tool diameter
End of evaluation grid
End

A.5 Tool Path Planning for Flank Milling




1
Evaluate maximum curvature at second curve
Evaluate maximum curvature at first curve
Maximum curvature=max(cur 1,cur2)
Create Tool Model
Maximum tool diameter Minimum tool length
Evaluate workpiece bounding box
At each evaluation position
Create normal vector
Position tool tangent to the surface points
Evaluate deviation distance along tool contact line
Get maximum deviation distance and location
Tolerance < maximum deviation
FALSE
Generate tool path
TRUE
End