VIRTUAL TURNING SYSTEM

by

JING ZHOU

B.Sc., Zhejiang University, Hangzhou, China, 1993

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES
(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

June 2005

Abstract

The goal of machining industry is to produce the first part correctly and most optimally without resorting costly trials on the shop floor. This thesis presents a Virtual Turning system which predicts the physics of machining rotational parts before actual production on the shop floor. As opposed to measurement of physical dimensions, cutting forces, torque, and power, they are predicted in virtual environment by integrating the laws of metal cutting process and the geometric and solid modeling of the tool-workpiece engagements along the tool path.

The proposed Virtual Turning has two fundamental modules. The first module identifies the tool–workpiece engagement geometry along the path, which is used by the second, cutting process simulation engine.

The initial workpiece geometry and tool path (Cutter Location) are imported from commercial CAD/CAM systems using industry standard IGES or STEP NC graphics formats. The tool-workpiece intersections along the tool path are identified by applying Boolean intersections of the two parts represented by their Boundaries. In order to expedite the time consuming computations, the in-process machining features along the path are classified, engagement conditions are parametrically modeled, and recalled instead of using Boolean operations recurrently along the tool path. The proposed hybrid model which consist of tool-workpiece engagements modeled by features or solid to solid intersections, can handle turning of a verity of two dimensional, symmetric rotational parts.

The contact length between the cutting edge and workpiece, and the chip area removed at each tool position are calculated by applying Green's Theorem to the tool-workpiece engagement boundary. The cutting force coefficients are modeled as a function of chip area, cutting edge contact length, tool geometry, feedrate and cutting speed. The cutting forces, torque, power and static deflections of the tool on the finish surface are predicted along the tool path. The algorithm can handle variety of tool motions which include taper and contour turning operations.

The two dimensional Virtual Turning system is experimentally validated in machining a sample shaft with circular and taper features.

Table of Contents

List of Tables

List of Figures

Acknowledgement

I would like to express the deepest appreciation to my research supervisor Dr.Yusuf Altintas for his valuable instruction, guidance, support, understanding and patience, which he has provided throughout my research at University of British Columbia. I would also like to extend my deepest gratitude to my co-supervisor Dr. Derek Yip-Hoi. He has taken an enormous amount of effort to instruct and help me about the academic and language matters. Without his guidance and persistent help this research would not have been possible.

I wish to thank all my colleagues in the Manufacturing Automation Laboratory for sharing with me their knowledge and experience, especially to Fuat and Dimitri, they have given me numerous suggestions when I had problems. And also Fuat helped me finish lots of machining experiments, which are very important to my research. I have learned a lot of them. I would like to thank Xuemei, Joseph, and Xiaobo, they gave me many helps in solid modeling. I also want to thank Yuzhong, a truthfully friend, he helped me in many ways.

Finally, words alone cannot express the thanks I owe to Lifeng, my husband, for his persistent encouragement, assistance and patient; to my mother Shumin and my father Rongxian, for their lifelong love and unwavering support; and to my son Ricky, for him I can overcome any difficulty. This thesis and my all previous success are dedicated to them.

Nomenclature

A	uncut chip area of the entire engagement
A_1	uncut chip area of region 1
A_2	uncut chip area of region 2
b	width of cut
C_i	tool center position at i^{th} feed step on a toolpath
C_{i-1}	tool center position at $i\text{-}1^{th}$ feed step on a toolpath
CL	cutter location
d	depth of cut (*mm*)
f	feedrate (*mm/rev*)
fiF	form invariant feature
F_t	tangent force
F_r	radial force
F_f	feed force
F_{fr}	friction force
F_{frc1}	friction force in region 1 of an uncut chip area
F_{frc2}	friction force in region 2 of an uncut chip area
$\mathrm{F_t}$	global tangent force
$\mathrm{F_r}$	global radial force
$\mathrm{F_f}$	global feed force
$\mathrm{F_x}$	cutting force in X axis direction
$\mathrm{F_y}$	cutting force in Y axis direction
$\mathrm{F_z}$	cutting force in Z axis direction
G_1	gravity center of region 1
giF	geometric invariant feature
h	chip thickness
ipF	in-process feature

K_{tc}	cutting coefficient of tangent force
K_{rc}	cutting coefficient of radial force
K_{fc}	cutting coefficient of feed force
K_{te}	edge coefficient of tangent force
K_{re}	edge coefficient of radial force
K_{fe}	edge coefficient of feed force
K_{frc1}	cutting coefficient of friction force in region 1
K_{frc2}	cutting coefficient of friction force in region 2
K_{fre}	edge coefficient of friction force
l	distance between C_i and C_{i-1}
l_1	the distance from P_1 to V
l_2	distance from P_2 to V
L_c	chip-cutting edge contact length
L_{c1}	chip-cutting edge contact length of region 1
L_{c2}	chip-cutting edge contact length of region 2
MRA	material removal area
mrF	in-cut material removal features
O	circular toolpath center position
P_a	tool nose arc edge upper tangent point
P_b	tool nose arc edge lower tangent point
P_c	tool end cutting edge lower right point
P_d	tool side cutting edge upper left point
P_1	tool nose curve upper tangent point of a tool
P_2	intersection point between cutting edges of two tools
P_3	Intersection point between P_1C_i and the tool centered at C_{i-1}
Q_i	workpiece boundary position intersected with current tool
Q_{i-1}	workpiece boundary position intersected with previous tool
Q	the center position of a circular workpiece boundary edge

R	circular toolpath radius
R_1	curve region of an uncut chip area
R_2	Polygonal region of an uncut chip area
r_ε	tool nose radius
r_q	the radius of a circular workpiece boundary edge
S	the length of a toolpath
teF	tool engagement features
trF	transient feature
TD	boolean difference between two tools
TWE	tool-workpiece engagement
TSA	tool swept area
$Toolpath$	a tool path
$Toolpaths$	tool paths
T_s	start position of a toolpath
T_e	end position of a toolpath
$\vec{V}$	instantaneous feed direction at tool contact point
V	cutting speed (m/s)
α	toolpath angle
β_a	friction angle (degree)
ψ_r	side cutting edge angle
ψ_r'	equivalent side cutting edge angle with respect to feed
κ_r	end cutting edge angle
ϕ_L	effective lead angle
ϕ_1	effective lead angle of region 1
ϕ_c	shear angle (degree)
τ_s	shear stress (N/mm^2)

Chapter 1

Introduction

The manufacturing of shafts, gears, discs and family of all rotational components involves turning operations. The rotational parts typically have varying geometric features along the axis with discontinuities, such as slots, keyways, and grooves. In general, all rotating mechanical parts used in common machinery, such as shafts and gears used in gear boxes, automobile engines, aircraft engine gas turbines, are produced with turning operations. The aim of manufacturing engineers is to optimize the machining cycle time of turning operations while respecting process constraints such as torque and power limits of the machine, breakage of the tool, dimensional tolerance of the part and chatter vibration limits of the machine tool and workpiece structures. The process constraints can be respected by selecting suitable feed, speed, depth of cut and tool geometry. However, the present practice in industry is based on the past experience of process planners only, and the selected cutting conditions may either be too conservative for high productivity machining, or too aggressive which leads to failure and repeated trials until satisfactory performance is achieved.

The objective of this thesis is to create foundations of a virtual turning system which is capable of predicting the process behavior before any actual turning test is conducted on real machines. In addition, the virtual turning should lead to optimization of feeds and speeds which lead to minimum machining cycle time, i.e. high productivity, while respecting the physical limits of the process and machine tool.

The virtual turning can be realized by modeling the tool-workpiece intersection geometry along the toolpath, and modeling the mechanics of turning which leads to realistic prediction of cutting forces, torque, power, deflections and vibrations. Most of the past research has either focused on the modeling of basic cutting mechanics and dynamics of turning process, or geometric modeling of material removal process in solid modeling environment. There has not been much research activity in realizing an integrated virtual machining environment which includes the physics of the process.

Virtual turning system has tool–workpiece engagement identification and modeling of process mechanics as a function of tool–workpiece engagement, tool geometry, feed, depth and speed of the cut. Further, optimization of process variables as a function of physical limits of the machine and cutting tool can be achieved by exploiting their results. The thesis presents research conducted in the aforementioned subjects, and their integration to achieve virtual turning system.

Henceforth, the thesis is organized as follows.

Chapter 2 reviews the relevant previous work in the fields of solid modeling of tool-workpiece intersection and mechanics of turning. The current CAD/CAM systems do not have any built-in algorithm which provides the tool-workpiece intersection. The relevant literature in extracting such geometric information along the toolpath is reviewed. The tool-workpiece engagement geometry may continuously vary along the toolpath, and it strongly affects the uncut chip area, hence the resulting force amplitudes and directions vary at each feedrate increment. The computational cost and accuracy of chip and force calculation methodologies reported in the literature are presented.

Chapter 3 provides architecture of the proposed Virtual Turning system. The system consists of two main modules: The first module identifies tool-workpiece engagement conditions and chip geometry at discrete tool motion intervals, which are used to predict cutting forces, torque, power and deflections in the second module. The inputs and outputs, solid modeling techniques, the influence of the feed motion direction in contour turning, and the assumptions of the system are presented.

Chapter 4 describes the algorithms developed to identify tool-workpiece intersection in turning operations. The workpiece geometry and NC Tool Path, i.e. Cutter Location (CL) file are imported from standard CAD/CAM platforms using IGES or STEP NC standards. The intersection of tool and workpiece is identified by two new techniques. The first method is based on the intersection of solid models of workpiece and tool on ACIS solid modeling kernel. The computational cost and robustness of the pure solid modeling approach led to the development of the second algorithm which integrates both solid model and feature-based engagement methods. The chip area is then predicted by applying Green's Theorem to the identified tool-workpiece intersection conditions.

The prediction of cutting force, torque and power is presented in Chapter 5. The previously reported mechanistic model of the turning process is adopted by considering changing chip area and orientation of the cutting forces along the cut. The extensions to the algorithm allow handling of contour turning operations at discrete feed increments. The proposed Virtual Turning system is experimentally validated in machining a sample shaft with varying geometry.

The thesis is concluded in Chapter 6. The contributions to the literature in Virtual Turning are summarized and the future research directions which lead to the handling of arbitrary tool and workpiece profiles are discussed.

Chapter 2

Literature Review

2.1 Introduction

Turning is one of the most commonly used metal cutting operations in industry. Many research projects have focused on the cutting mechanics, modeling and simulation of turning to understand the physics of the process and increase its efficiency. The modern cutting-process models have stemmed from a fairly good understanding of the metal cutting process gained through the experimental findings of the early years of machining research. Part of this literature review presented in this chapter is concerned with the cutting mechanics, uncut chip area, and chip thickness.

The proposed Virtual Turning system, i.e., the geometric and physical simulation of the turning process, is based on the integration within one system of geometric and solid modeling models and static force prediction models for different types of turning process. The system thus combines components in the areas of mechanics of turning, engagement geometry calculation, solid modeler techniques, feature recognition methods, and swept volume generation algorithms. A literature survey related to these aspects is presented in this chapter.

2.2 Force Prediction Models

The cutting forces in turning operations are typically represented by the three orthogonal force components, namely tangential F_t, radial F_r, and feed F_f forces. These forces are proportional to the area of the interference between the tool and the workpiece (uncut chip area) as well as the length of engagement between the tool edge and the workpiece, and can be calculated as [Altintas, 2000] and [Armarego et al., 1985]:

$$\begin{aligned} F_t &= K_{tc}A + K_{te}L_c \\ F_r &= K_{rc}A + K_{re}L_c \\ F_f &= K_{fc}A + K_{fe}L_c \end{aligned} \tag{2.1}$$

If the tool rake face has an irregular geometry due to chip breaking grooves and chip tool contact restriction features, the cutting coefficients are identified using mechanistic models. A series of cutting tests are conducted with the specific tool at different speeds, radial depth of cuts, and feedrates. The coefficients are evaluated by curve fitting the force expressions to the measured cutting forces and chip geometry.

If the rake face of the tool is smooth and uniform, it is possible to model the cutting edge as an assembly of oblique cutting edges [4,7]. The cutting pressure at each discrete oblique cutting edge element is modeled by applying the orthogonal to oblique transformation method proposed by Armarego [7]. Both approaches will be introduced in this literature review.

2.2.1 Orthogonal to Oblique Transformation for Corner-radius Tools

If the insert's rake face is uniformly flat without chip breaking or contact reduction grooves, the turning insert's curve cutting edge can be considered an assembly of oblique cutting edge elements. Oblique cutting mechanics laws lead to the prediction of cutting pressure at each discrete cutting edge element, which depends on the discrete chip area, edge geometry, and orthogonal cutting parameters of the work material (i.e., shear stress, shear angle and friction angle) which are mapped using classical mechanics laws proposed by Armarego [7]. The details of the orthogonal to oblique cutting transformation can be found in [1,2,4,7].

The three cutting force components can be expressed as follows:

$$\begin{aligned}
F_t &= F_{tc} + F_{te} = K_{tc} \cdot b \cdot h + K_{te} \cdot b \\
F_f &= F_{fc} + F_{fe} = K_{fc} \cdot b \cdot h + K_{fe} \cdot b \\
F_r &= F_{rc} + F_{re} = K_{rc} \cdot b \cdot h + K_{re} \cdot b
\end{aligned} \tag{2.2}$$

where the oblique cutting coefficients are presented as follows:

$$\begin{aligned}
K_{tc} &= \frac{\tau_s}{\sin\phi_n} \cdot \frac{\cos(\beta_n - \alpha_n) + \tan i \tan\eta \sin\beta_n}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2\eta \sin^2\beta_n}} \\
K_{fc} &= \frac{\tau_s}{\sin\phi_n \cos i} \cdot \frac{\sin(\beta_n - \alpha_n)}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2\eta \sin^2\beta_n}} \\
K_{rc} &= \frac{\tau_s}{\sin\phi_n} \cdot \frac{\cos(\beta_n - \alpha_n)\tan i + \tan\eta \sin\beta_n}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2\eta \sin^2\beta_n}}
\end{aligned} \tag{2.3}$$

The shear stress (τ_s), the shear angle (ϕ_n), and the friction angle (β_n) are determined from the results of the orthogonal cutting tests [1,2]. The uncut chip area is divided into three regions (Figure 2.1) due to the tool nose curve and lead cutting angle.

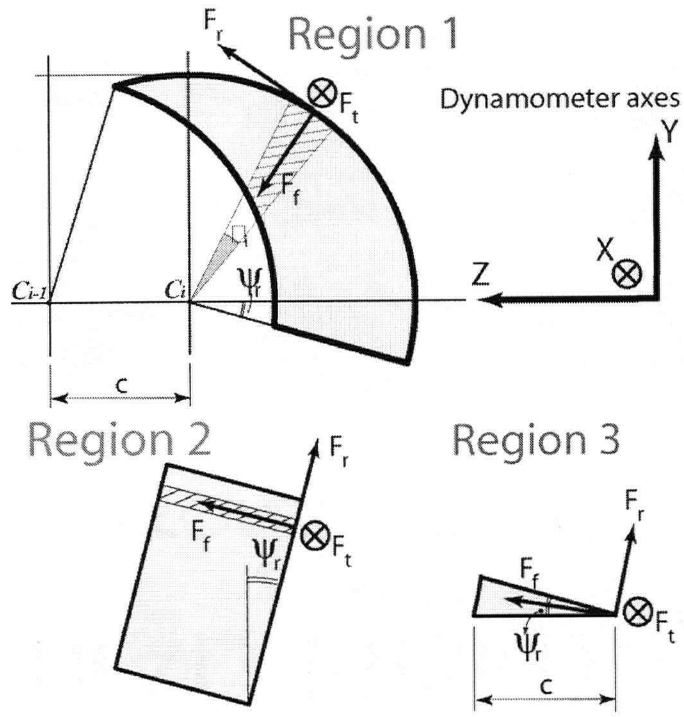


Figure 2.1: Orthogonal to Oblique Transformation for Corner-radius Tool

In region 1 the uncut chip area is divided into small differential elements, for each of these elements, the oblique tangential, radial, and feed forces can be determined as:

$$\begin{aligned} F_{t,i} &= K_{tc,i} \cdot A_{1,i} + K_{te} \cdot L_{c,i} \\ F_{r,i} &= K_{rc,i} \cdot A_{1,i} + K_{re} \cdot L_{c,i} \\ F_{f,i} &= K_{fc,i} \cdot A_{1,i} + K_{fe} \cdot L_{c,i} \end{aligned} \tag{2.4}$$

Where, $A_{1,i}$ is the chip area of i$^{\text{th}}$ element, and $L_{c,i}$ is the chip-cutting edge contact length of i$^{\text{th}}$ element. By summing all the respective force components, the cutting force in region 1 can be determined as equation (2.5), in which θ_i is the uniform angular increment of each element:

$$
\begin{aligned}
F_{x,1} &= \sum_{i=1}^{n} F_{t1,i} \\
F_{y,1} &= \sum_{i=1}^{n} (F_{f1,i} \sin\theta_i - F_{r1,i} \cos\theta_i) \\
F_{z,1} &= \sum_{i=1}^{n} (F_{f1,i} \cos\theta_i - F_{r1,i} \sin\theta_i)
\end{aligned}
\tag{2.5}
$$

In region 2 and region 3, the approach angle is assumed to be the side cutting edge angle (ψ_r) and half of this angle respectively. The cutting force components can be calculated as:

$$
\begin{aligned}
F_{x,2} &= F_{t2} \\
F_{y,2} &= F_{f2} \sin(-\psi_r) - F_{r2} \cos(-\psi_r) \\
F_{z,2} &= F_{f2} \cos(-\psi_r) - F_{r2} \sin(-\psi_r)
\end{aligned}
\tag{2.6}
$$

and

$$
\begin{aligned}
F_{x,3} &= F_{t3} \\
F_{y,3} &= F_{f3} \sin(-\psi_r/2) - F_{r3} \cos(-\psi_r/2) \\
F_{z,3} &= F_{f3} \cos(-\psi_r/2) - F_{r3} \sin(-\psi_r/2)
\end{aligned}
\tag{2.7}
$$

The total forces for the entire uncut chip area in global X, Y and Z directions are found as follows:

$$
\begin{aligned}
F_x &= F_{x,1} + F_{x,2} + F_{x,3} \\
F_y &= F_{y,1} + F_{y,2} + F_{y,3} \\
F_z &= F_{z,1} + F_{y,2} + F_{z,3}
\end{aligned}
\tag{2.8}
$$

The advantages of this model are that it is valid for a range of cutting tool geometries and have been verified for good force prediction from past research. The disadvantages are that the tool nose curve needs to be discretized into small segments at each feed step, because this model is only valid for straight cutting edges. Also the cutting coefficient expressions are complicated, and they have to be evaluated for each element. Differential cutting forces are summed up to the total XYZ forces, but these XYZ forces predicted only represent for one feed step. These disadvantages make the orthogonal to oblique transformation approach less efficient in modeling the contour turning, which likely has large numbers of feed steps. Moreover, this model is valid only for the tools with a flat rake face, which limits the model to be used for general turning processes. Therefore, a mechanistic model is adopted in this research, which is described in the next section.

2.2.2 Mechanistic Force Model for Corner-radius tools

In the mechanistic model proposed by Atabey et al. [1,2], cutting forces are represented by a tangential component *(F_t)* and a frictional component *(F_{fr})* as shown in Figure 2.2. F_{fr} is further resolved into radial (F_r) and feed (F_f) forces. Cutting forces at each feed step are modeled as a function of the uncut chip area *(A)* and the chip-cutting edge contact length *(L_c)* as shown in equation (2.9):

$$\begin{aligned} F_t &= F_{tc} + F_{te} = K_{tc} \cdot A + K_{te} \cdot L_c \\ F_{fr} &= F_{frc} + F_{fre} = K_{frc} \cdot A + K_{fre} \cdot L_c \end{aligned} \tag{2.9}$$

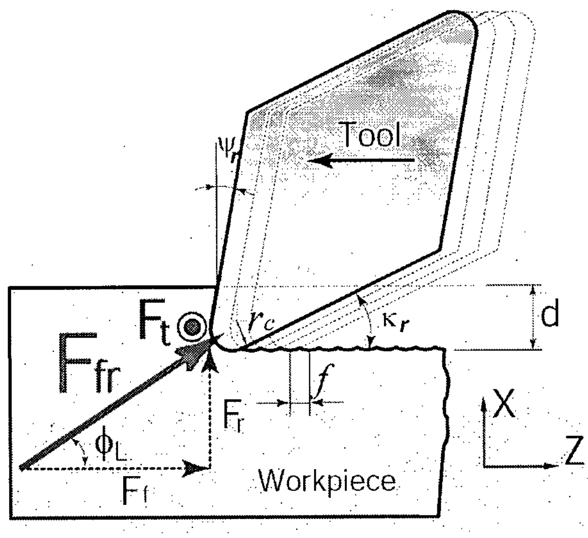


Figure 2.2: Mechanistic Force Model

To account for the differences in cutting mechanics between the tool's leading edge and nose regions, an approximated polygonal region is used for force prediction along the leading edge and a region bounded by arcs and lines is used to account for the nose. The continuously changing oblique angle at the nose is the reason for the differences. As a result, the uncut chip area is divided into two zones as illustrated in Figure 2.3.

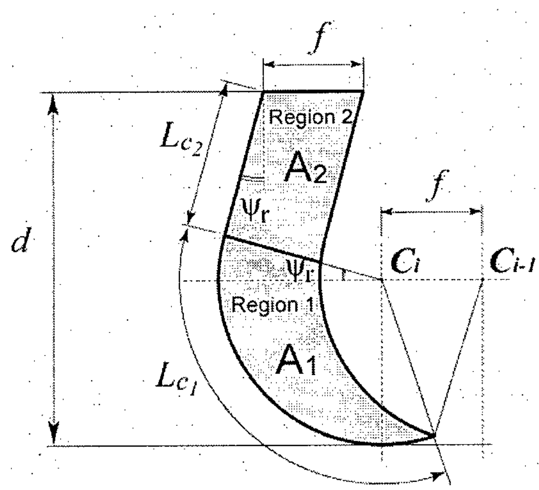


Figure 2.3: Uncut Chip Area Decomposition

The frictional force in equation (2.9) is modified as follows to account for this change:

$$F_{fr} = K_{frc1}A_1 + K_{frc2}A_2 + K_{fre}L_c \tag{2.10}$$

Where, the areas (A_1 and A_2) are corresponding to the areas of two regions. K_{tc}, K_{frc1}, and K_{frc2} are empirical proportionality coefficients between the corresponding cutting forces and the uncut chip area, they are modeled as a nonlinear function of cutting speed and uncut chip area or chip-cutting edge contact length; K_{te} and K_{fre} are empirical proportionality coefficients between the corresponding edge forces and the length of the tool cutting edge engaged with the workpiece.

For a given tool-workpiece material combination and tool geometry, the cutting coefficients can be identified from a set of cutting tests where the forces are measured for ranges of feeds (f) and depths of cut (d), and the collected data is processed with multivariable regression analysis [Altintas, 2000]. Later, the tangential force and friction force are resolved to XYZ forces by using an effective lead angle. With all necessary inputs described above, the instantaneous force components can be calculated from the above equations at each feed step of the cutter.

Rohi G. Reddy [29,30] used normal force (F_n) and the frictional force (F_f) in his mechanistic model for contour turning as shown in equation (2.11). Similarly, these two forces are resolved to the global X, Y, Z direction at the end.

$$\begin{aligned} F_n &= K_n \cdot A_c \\ F_f &= K_f \cdot A_c \end{aligned} \tag{2.11}$$

Where K_n and K_f are specific cutting energy coefficients, which are functions of equivalent chip thickness, cutting velocity, and normal rake angle. It can be seen that this force model is in fact a simplified mechanistic force model analogous to equation (2.9) for the static case, but the uncut chip area A and cutting coefficients have different definitions.

From the review of research up-to-date, it can been seen that the turning static forces can be predicted using the well known existing force models, in which forces are proportional to the area of the interference between tool and workpiece and the cutting coefficients. Therefore, only the mechanistic force model proposed by Atebey [1,2] is adopted in this research. However, it is necessary to develop a new method to identify the continuously changing tool workpiece intersections and the cutting coefficients effectively and accurately. Some developed methods to predict chip geometry are reviewed in the next section.

2.3 Prediction of Chip Geometry

Since the early days of metal cutting research, researchers have observed the machining force to be proportional to the cross-sectional area of the uncut chip being removed. This *chip area* is defined as that area bounded by the tool edge profiles corresponding to two tool passes. The theoretical analysis of the machining processes, dating back to the early 1940s or before, has considered this fact and modeled the machining force components as proportional to the chip area. However, in the presence of vibration, runout, or special cutter designs such as contour turning applications, establishing an analytical chip-area expression is not a trivial matter when working with processes used by industry, i.e., contour turning, those that exhibit complex tool forms.

The most common tool form seen in turning and boring consists of a straight major lead cutting edge, a straight minor cutting edge, and an edge with a corner radius that connects /

blends together the two straight edges. Ozdoganlar [26] termed the ensuing chip-area representation and analysis *corner-radiused tools*.

The chip thickness is a calculated factor based on the chip area. The Equivalent chip thickness is used in calculating cutting forces and tool life, described 1936 by Woxen [9] as shown in Figure 2.4.

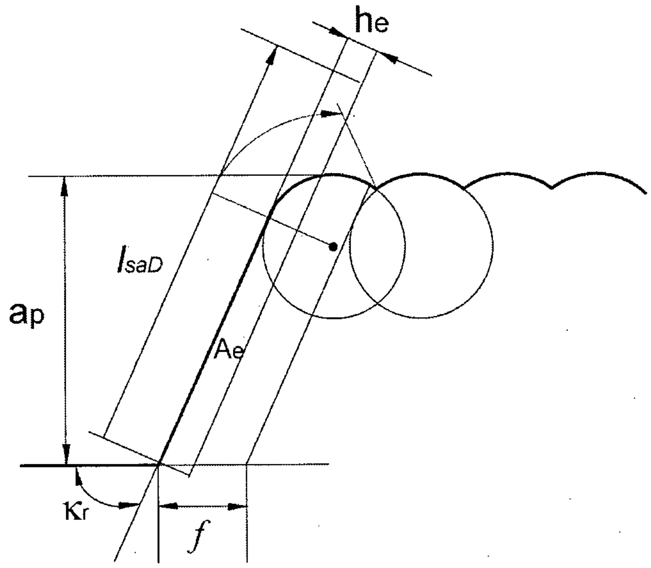


Figure 2.4: Equivalent Chip Thickness (h_e)

The equivalent chip thickness (h_e) is the quotient of the approximate chip area, which is the product of the depth of cut (a_p) and feedrate (f) divided by the active tool edge length (l_{saD}).

$$h_e = \frac{A_e}{l_{SaD}} = \frac{a_p \cdot f}{l_{SaD}} \tag{2.12}$$

where

$$l_{SaD} = \frac{a_p - r_\varepsilon \cdot (1 - \cos\kappa_r)}{\sin\kappa_r} + \frac{\kappa_r \cdot r_\varepsilon \cdot \pi}{180} + \frac{f}{2} \tag{2.13}$$

The active tool edge length l_{SaD} is considered as a straight line in the equivalent chip area, as presented in Figure 2.4. However, equations (2.12~2.13) are approximate values for the real chip thickness. Moreover, these geometric and trigonometric methods become complex when tool inclination and rake angle change.

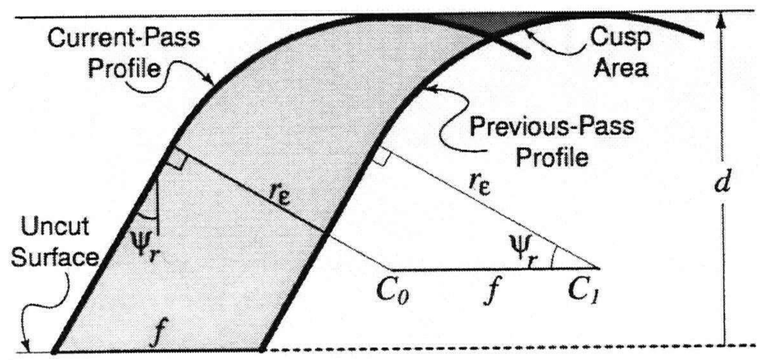


Figure 2.5: Simple Representation of the Corner-radiused Chip Area

A traditional graphical representation of the chip area is shown in Figure 2.5, where r_ε is the corner radius and ψ_r is the lead angle. For the simple situation shown here, where feed direction is defined to be parallel to the uncut surface and a depth direction is defined to be perpendicular to the feed direction, the commonly seen chip area expression is

$$a = fd - a_c \tag{2.14}$$

a_c is the area of the cusp left on the cut surface at the intersection of the current-pass and previous-pass profiles.

$$a_c = fr_\varepsilon - \frac{f}{2}\sqrt{r_\varepsilon^{\ 2} - \frac{f^2}{4}} - r_\varepsilon^{\ 2} \arcsin\left(\frac{f}{2r_\varepsilon}\right) \tag{2.15}$$

The area model is attractive, since *fd* is a simple product involving the depth of cut and feedrate, a_c is a fairly simple function of feed, corner radius and sometimes, in extreme cases, the lead and end-cutting edge angles. However, contour turning introduces profile-to-profile variations in the feed direction as well as the depth of cut value and direction. This representation cannot give the correct prediction.

A method for computing the exact chip area, in the presence of depth and feed variations, has been developed by Endres [29] based on the addition and subtraction of geometric shapes. Figure 2.6 illustrates the geometric shapes for the large depth case. Subtracting the crosshatched area, a circular segment, from the shaded area composed of three triangles and a circular segment, the exact area is obtained. However, this exact result is computationally complex and fairly algorithmic since it includes many cases with several conditions and requires the coordinates of each of the five points in Figure 2.6 to be computed. Moreover, this primary trigonometric method cannot give general expression of the chip geometry.

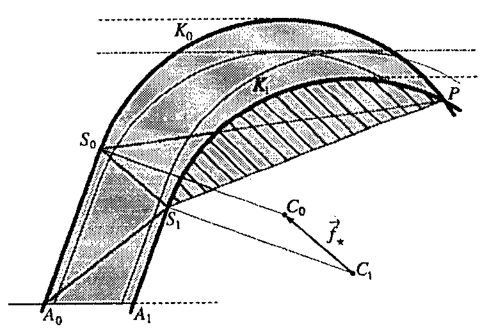


Figure 2.6: Exact Area Calculation Using Geometric Shapes

Ozdoganlar [26] proposed an analytical representation of the area which is employed for a tool at each time step in time-domain simulations. This analytical representation also opens the door to analytical machining dynamics, where one seeks analytical solutions for stability limit and vibration level.

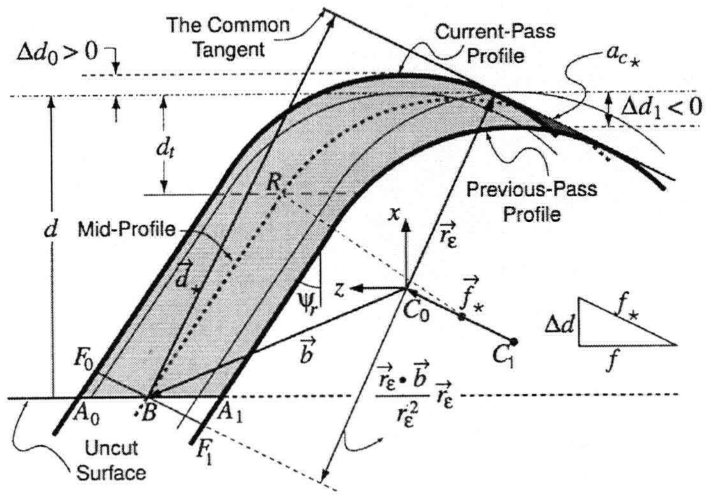


Figure 2.7: Chip-Area Geometry with a Depth-Direction Variation

Chip-area expressions are derived for "small" and "large" depth cases using a vectorial approach. Large depth of cut case is shown in Figure 2.7.

Large depth of cut $$a_l = fd_m + r_\varepsilon(f_* - f) + \Delta d\left(\frac{r_\varepsilon(1-\sin(\psi_r))}{\cos(\psi_r)} + d_m \tan(\psi_r)\right) - ac_* \qquad (2.16)$$

Small depth of cut $$a_s = fd_m + r_\varepsilon(f_* - f) + \Delta d\sqrt{d_m(2r_\varepsilon - d_m)} - a_{c*} \qquad (2.17)$$

Where $d_m = \frac{(d_0 + d_1)}{2}$ is the mean depth of cut. f_*, a_{c*} is the equivalent feed and cusp area. Since the presented chip area is an approximation, it involves analytical error compensation and numerical error compensation. Therefore, it is not desirable for static force prediction.

Another analytical chip load solution has been developed by Rohit [29,30]. The approach is to divide the intersection region into three zones, depending upon the tool parameters and the cutting conditions, and to calculate separately the portion of the chip load in each zone, as depicted in Figure 2.8.

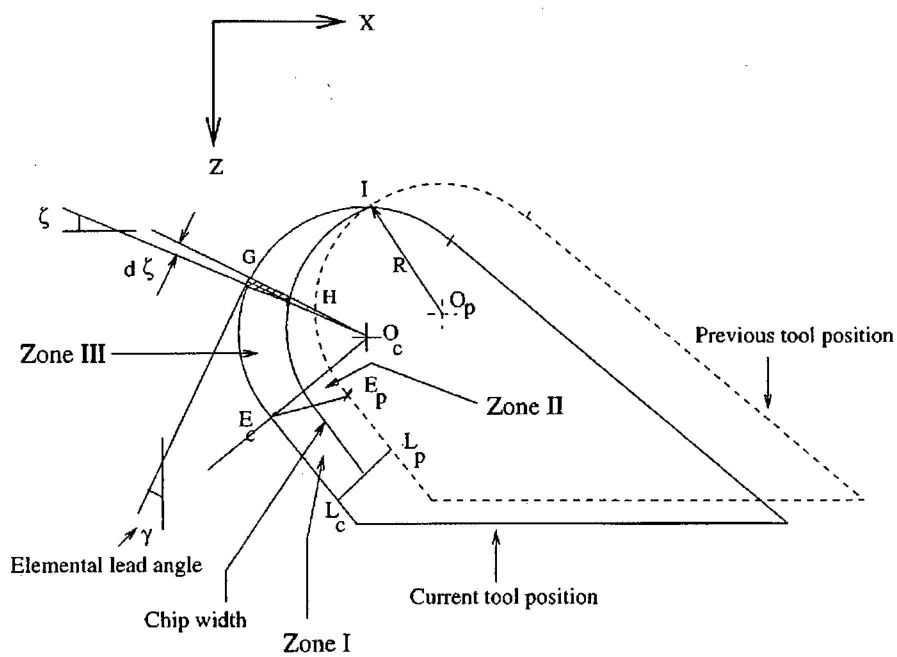


Figure 2.8: Chip-Area Geometry

The area of Zone 1 $\quad A_{cI} = area\ of\ quadrilateral\ L_cL_pE_pE_c$ (2.18a)

The area of Zone 2 $\quad A_{cII} = \int_{\xi_{Ep}}^{\xi_{Ec}} \left(\frac{O_cG^2 - O_cH^2}{2} \right) d\xi$ (2.18b)

The area of Zone 3 $\quad A_{cIII} = \int_{\xi_{Ec}}^{\xi_1} \left(\frac{R^2 - OcH^2}{2} \right) d\xi$ (2.18c)

The total uncut chip area is

$$A_c = A_{cI} + A_{cII} + A_{cIII} \tag{2.19}$$

This model involves numerical integration, and an equivalent chip thickness is calculated for identifying the cutting coefficients. A number of calculations are required to separate the uncut chip area to three zones and get the chip areas, and this method divides the plane of the tool motion into four quadrants, additional identifications are also required to cast the cutting conditions into one of these four quadrants. Instead, a general analytical area representation for all cases is desirable.

Atabey [1,2] obtained the uncut chip area by summing up the contributions from a discretization of the intersection into approximate geometric elements. See Figure 2.9.

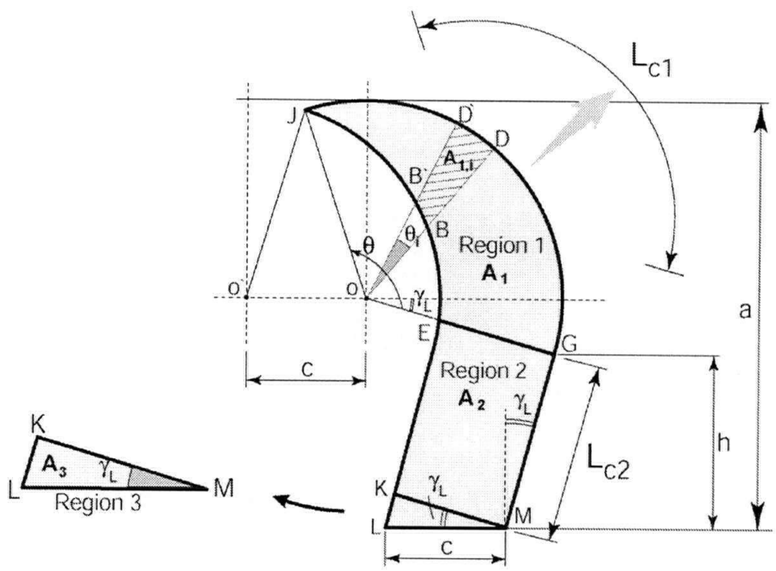


Figure 2.9: Uncut Chip Area Calculations from Elements

The total chip area in Region 1 is evaluated by a discrete summation of all differential elements in the curved region. Region 2 is considered to be a rectangle, although one side of it has a slight curve caused by the corner radius of the previous tool position. Region 3 is a simple triangle. Finally, total uncut chip area is found by adding the areas for each region. This method also uses numerical integration, and the area is approximate. This method can be used in one constant cutting condition, but it is not efficient for continuously changing geometry cases.

Armarego [7,32] identified nine different types of tool-workpiece intersections in turning depending on the depth of cut and feedrate limits, and developed the analytical solutions. But the cutting conditions were originally developed for longitudinal straight cuts. If this method is used on contour turning, it is difficult to cast an uncut chip area to those 9 categories due to the complex calculations of three varying feedrate limits along the contour toolpath. Also it approximates the workpiece boundary as a straight line. Since the feedrate in the turning operation is usually smaller than nose diameter, there are three cases in those nine cases that will not be considered in this research.

Other significant research has developed area calculation methods for tools with curved geometry, but most of them are approximations and cannot capture the varying geometries along the arbitrary toolpath and workpiece surface.

In this thesis, taking advantage of Green's Theorem-based analytical area calculation, the intersection area and other parameters, which are represented by simple algebraic integration formulae, can be calculated accurately and effectively. Moreover, this analytical solution is generalized for any arbitrary closed area, and the boundary conditions are easy to obtain by calculating few intersection points between the line and arc. The details are given in Chapter 4.

2.4 Solid Modeler and Z buffer Methodology

Area calculation methods investigated in the previous section are in fact not capable of capturing the tool-workpiece engagement along the whole arbitrary toolpath and in-process workpiece, since the instantaneous intersection geometry varies dynamically and hard to predict. Some research that has been done for contour turning requires the workpiece must be simple so that it cannot be used widely.

The biggest difficulty is that the depth of cut varies depending on the relative distance and direction between a toolpath and an in-process workpiece boundary. The accurate depth of cut can be identified by using solid modeling in which the workpiece and the tool are represented as B-rep (Boundary Representation) models, and their in-process geometries and topologies are obtained by applying Boolean operations with the swept area of the tool. In this research a Solid Modeler is used as the basis for extracting the intersections from the continuously changing geometries.

Researchers have in the past investigated the potential of solid modelers to support modeling machining processes [6,9,35]. Computational complexity was identified as one of the difficulties in adopting this approach. In addition, research has also demonstrated geometric and solid modeler cutter-workpiece intersection calculations within the context of an integrated virtual machining environment where the modeler provides inputs to the process models [5,6,13,14]. While results are promising, they do not address all the possibilities that can come from the range of geometry, processes, cutting tools, and machine tool axis configurations that are encountered. In particular, most approaches are for 2½ D milling operations. There is little research that focuses on turning operations, especially when the initial workpiece is non-cylindrical and when multiple turning operations (facing, profiling, grooving) lead to intersecting machining features. The proposed research partly addresses this deficiency. It is also important to mention that other techniques have been studied for finding tool workpiece engagements. Most notable is the Z-buffer method originally developed for NC verification but adopted to obtain engagement geometry. Examples include Takata [35], Jerard [15,16] and Lazogolu [23]. However, again these researchers target end mill part intersections, but not turning.

In this research, ACIS[1] Solid Modeler kernel is used to model the turning process, where the in-process workpiece and tool are constructed as B-rep solid models, tool travels along tool paths, and Boolean operation is used to subtract the intersection area and update the workpiece after each toolpath.

2.5 Feature Recognition Technologies

Significant research has investigated the problem of feature recognition. Reviews of this research can be found in [33,36]. Feature recognition addresses the problem of identifying engineering relevant regions of interest (faces, edges, points) from a CAD model.

Typically, two main approaches have been used for recognition of 2D rotational features. One is syntactic pattern recognition, used by Jakubowski(1980), Srinivason (1985), and Li (1988). The 2D boundary of the part is captured as a string of geometric primitives that are then parsed using a grammar to identify feature patterns. This approach does not consider the workpiece boundary, and additional steps are required to generate machining volumes. Another technique is the rule-based feature recognition approach, which was used by Davies et al. (1988) and Joseph and Davies (1990). Features are recognized using decision logic expressed as rules within an expert system as part of a "backward planning" strategy.

---

[1] In this research the ACIS solid modeler product is used to model and manipulate geometry

Feature recognition research has directly supported Computer-Aided Process Planning by targeting features on the final part geometry. The surfaces of these features are used to identify appropriate machining operations to be applied to the initial workpiece. Identifying features for supporting process modeling has not received significant attention. These features differ in that they appear on in-process states of the workpiece. This new type of feature is the focus of the technique described in this thesis. The feature identification approach used is based on a 2D area decomposition algorithm proposed by Cho et al. (1994) and Sakurai and Chin (1994). The difference of in the approach used in this research is the type of decomposition used, and the definitions of in-process machining features that are specified for turning operations. Since there are a small numbers of engagement conditions in turning operations, feature identification method is developed to extract the intersection geometry from three machining features to enhance the computational efficiency.

2.6 Swept Volume Techniques

Significant amounts of research have focused on developing swept volume algorithms since swept volumes are used in a variety of applications such as robotic analysis, collision detection, machining verification, and simulation. These methodologies can be classified into mathematical approaches and engineering approaches.

Examples of mathematical approaches reported are the Jacobian Rank Deficiency method (JRD) and Sweep Differential Equation (SDE) approach. JRD method has only been demonstrated in parametric and implicit surface sweeping with multiple parameters. The SDE method has been demonstrated for planar parametric curves sweeping. These general methods have provided some well-established solutions for analytical curves and surfaces sweeping. However, these approaches are not practical in solid model-based applications because analytical expressions of curves or surfaces are not always available.

Some engineering methods have been developed to generate swept volumes for NC verification for 5-axis machining. For instance, Sheltami et al. (1998) uses generating curves to get swept volumes of toroidal cutters, Roth et al. (2001) do surface swept by a toroidal cutter during 5-axis machining. Weinert et al. (2003) generates swept volume for the simulation of machining processes.

A swept area of a two-dimensional turning tool is the union of the area occupied by the tool at all positions during the motion. The swept area is generated using a boundary representation of the border of the sweeping body, such as line segments and arcs. The boundary of the swept area is developed as the envelope of all plane curves representing the boundary of the body at all positions of the body included in the sweep. Unfortunately, most planar moving bodies cannot be represented by a parametric equation as simple as a circle. The methodology for identifying "envelope points" is described in Ling and Chase [22].

In this research, since the toolpath and tool geometry are a combination of lines and arcs only, a simple swept area algorithm has been developed for the turning process, which is different from [22] in the way of finding extremal points and constructing edges. Details of the algorithms for linear toolpath and circular toolpath are explained in Chapter 4.

2.7 Summary

In this chapter, an outline of the literature in mechanics of turning, chip geometry calculation, solid modeler techniques, feature identification methods, and swept volume algorithms has been presented. It has been shown that well developed turning mechanics models can predict cutting forces accurately, as long as the chip geometry is provided correctly. The tool-workpiece intersection calculation becomes challenging along the contour turning, when the engagement changes dynamically. Solid Modeler techniques are used to model the whole turning process, while the computational complexity is a difficult issue. The contributions intended in this research have been placed in context with the reviewed literature.

Chapter 3

Overview of the Virtual Turning System

3.1 Introduction

The modern manufacturing sector requires rapid design, manufacturing, and deployment of products in small batch sizes. When the batch size is small and the lead period is short, the industry cannot afford to conduct costly test trials on the shop floor. The goal of virtual machining, as proposed by Altintas [CIRP 1991] is shown in Figure 3.1, is to machine the part using a mathematical model of the process in a simulation environment.

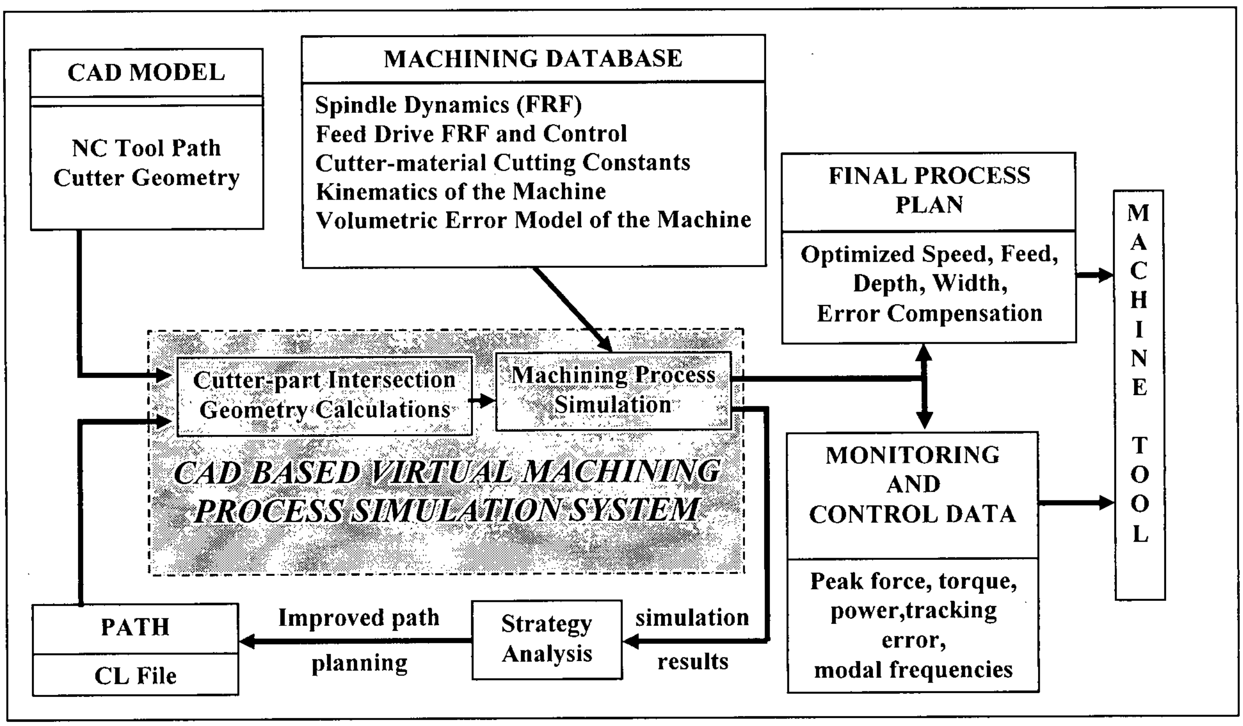


Figure 3.1: Virtual Machining Model Proposed by Altintas [CIRP 1991]

Altintas states that "The part must be produced accurately and most optimally in the shortest cycle period at the first trial on the shop floor, which is possible only if the mechanics of the metal-cutting process and the dynamic behaviour of the machine tool are modeled accurately using the laws of physics" [Altintas, NSERC–P&WC Industrial Research Chair Grant Application]. The proposed Virtual Turning system is a component of the CAD-based process simulation module in Virtual Machining Model proposed by Altintas [CIRP 1991] as shown in Figure 3.1, and developed in this thesis. This chapter provides the brief overview of this system, which includes two main modules and their inputs and outputs, the assumptions, and the capability of the system.

3.2 Overview of the Virtual Turning System

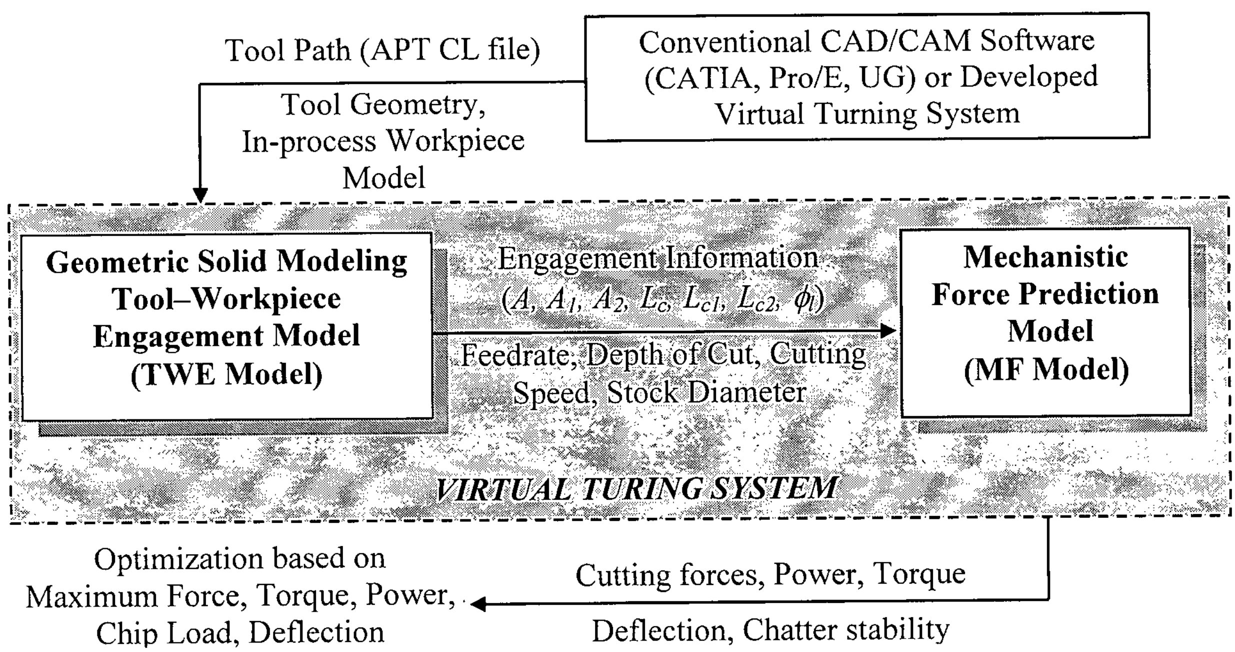


Figure 3.2: Virtual Turning System

Figure 3.2 shows a flowchart of the Virtual Turning system. This system is composed of two main modules, i.e., the Geometric and Solid Modeling Tool-Workpiece Engagement Model (*TWE* model), and the Mechanistic Force Prediction Model (*MF* model). In the *TWE* model, an intersection extracting methodology is developed. The outputs of the *TWE* model are the inputs of the *MF* model, where the cutting forces, power, and torque are calculated by using a mechanistic force prediction approach. The inputs and outputs of these two modules, along with the brief introduction of these modules are presented in the follows.

Three inputs are required to the *TWE* model. First, an APT CL file (Cutter Location file) generated from CAD/CAM software, which describes the toolpaths, is read and saved to a cutter location array. Each pair of nodes represents the start position and end position of one NC block in the APT file. If the toolpath is an arc, the center position of this arc, the radius, or the tangent direction of the start position are also stored in the data structure. In the simulation, the tool moves along the tool path, the instantaneous intersections and force calculations rely on the tool position and the feed direction at each machining step. Second, a 3D workpiece STEP file (Standard for Product Model Data file), which is also exported from CAD/CAM software, is translated to an ACIS[1] B-rep (Boundary Representation) model by a solid modeler translator, which is ready to be manipulated. Additionally, an in-process workpiece model, which is the updated workpiece or the final part after the intersection calculations in *TWE* model, can also be inputted as an initial workpiece for the next process simulation. Third, tool geometry, which is described by a nose radius, a side and an end cutting edge angles, and a side and a back rake

[1] In this research the ACIS solid modeler product is used to model and manipulate geometry

angles, are imported separately, since a turning CL file does not provide entire tool geometry information.

The intersections of the tool and the workpiece at every feed step along the toolpaths are identified using the *TWE* model. The outputs of the *TWE* model are uncut chip area (A, A_1, A_2 (mm^2)), chip-cutting edge contact length (L_c, L_{c1}, L_{c2} (mm)), feedrate (f (mm/rev)), depth of cut (d (mm)), cutting speed (V (m/min)), workpiece radius (r (mm)), effective lead angle (ϕ_l (rad)), and machining time (T (s)). These outputs are needed for force calculation in *MF* model. The instantaneous cutting forces, power, and torque, which are the outputs of the *MF* model, and also the workpiece deflection and chatter stability, which can be predicted from the forces easily, will be used to optimize the process at desired feed increments by selecting feeds and spindle speeds based on a set of machining constrains, such as machine tool maximum / minimum speeds and feeds, maximum power and torque, and stability limits.

3.3 Tool–Workpiece Engagement Model (*TWE* Model)

As described in the literature review, tool-workpiece engagement geometry can be predicted when the workpiece is simple, and when the tool trajectory, depth of cut and feedrate are known. However, for complex workpiece geometry and toolpath, the geometry of cut varies at every machining step and is hard to predict, one example is shown in Figure 3.3.

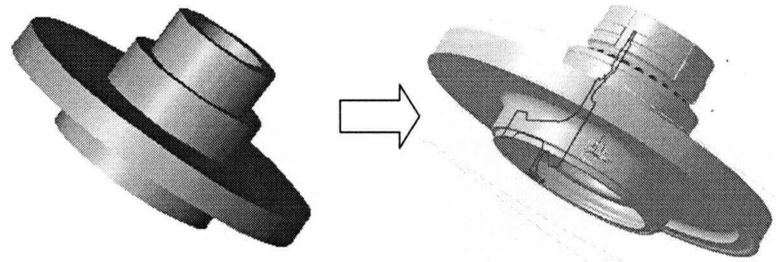


Figure 3.3: Dynamically Changing Engagement Geometry

One of the solutions to the problem of capturing and manipulating the realistic, complex geometry dynamically in the CAD/CAM environment is the use of solid modelers. The ACIS Solid Modeler is one of the most commonly used solid modeling kernels, thus the proposed *TWE* model in Virtual Turning system is based on the ACIS solid modeling kernel.

ACIS, the 3D Geometric Modeler, is an integrated software library of geometric and solid modeling algorithms which can be used in the development of any application requiring the representation and manipulation of 3D geometry. ACIS represents the exact shape of an object because it creates and records the equations of the curves and surfaces. A boundary representation (B-rep) is use to define complex 3D shapes in terms of the geometry of faces and edges and the topology (the relationships between these faces and edges) that define the physical boundary of the object.

In the machining process, the tool solid model moves along the toolpaths, and intersects the workpiece continuously. Boolean operations, which include union, subtraction, and intersection, are used to obtain the tool-workpiece intersection and update the in-process workpiece. Within these operators, first, all the intersections between the two bodies are identified. Second, the intersection graph is imprinted onto both bodies. This splits faces with these intersection curves into new faces. Third, a decision is made to determine which of the new faces are to be kept and which should be discarded. Finally, the new B-rep model after the Boolean operation is reorganized to ensure a valid topology.

Every Boolean operation leads to complex computations on the B-rep model, and the computation is not fully reliable. Due to the computational complexity and robustness problems surrounding Boolean operations, analytical and feature-based methodologies are developed in this research to increase the efficiency and robustness of the process simulation, which will be described in Chapter 4.

3.4 Mechanistic Force Prediction Model

In the force prediction model, a mechanistic approach, with the tool having a nose radius, is used to predict the cutting forces [1,2], which will be described in Chapter 5 in detail. As shown in the literature review, the cutting forces are represented as a function of the intersection geometries as shown in the follows:

$$F = K \cdot (A, L_c) \tag{3.1}$$

The intersection geometries come from *TWE* model, which is described in the previous section. Since the proposed Virtual Turning system is aiming to simulate the whole turning process, the *MF* model has the ability to predict forces, power and torque continuously, i.e., the *MF* model uses intersection geometries at every machining step to calculate the instantaneous forces along all cutting steps.

In contour turning, tool feed direction changes along the toolpath, while the forces are predicted with respect to the feed direction, i.e. the predicted feed force at each step is in the same direction with the instantaneous tool feed direction at that step, and the predicted radial force is in the direction that is perpendicular with the feed direction. These two local forces need to be resolved and summed in the global XYZ directions for further simulation and optimization.

Another challenge of this force model is to predict forces correctly and continuously for different types of turning. Since the mechanics of these different turning operations are regarded as the same cutting principles, one force model is applied for many types of turning operations. Therefore, no matter what type of a turning operation is, the predicted forces are carried out in the local coordinate system with respect to the cutting mechanics, and then these forces are projected onto the global XYZ directions for predicting the cutting forces, power and torque along the whole machining process.

After the cutting forces, powers and torques of the whole process are predicted, they can be presented together to find the critical process parameters, such as the maximum cutting forces, the maximum power and torque, and the maximum chip load (engagement area), compared with the machining constrains, such as the machine-allowed forces, power, torque, and chip load, to increase the material removal rate (increase feed or speed) if the process parameters are too low, or decrease the feed or speed if they are too high. And also the chatter stability is predicted by using depth of cuts (also form *TWE* model) and cutting speed.

3.5 Assumptions of the Virtual Turning System

In this section the assumptions that are made in this research are outlined. These are based on the limitations of the force prediction model adopted and simplifications to the cutting tool geometry.

- *Rigid workpiece and cutting tool:* As such deflections due to flexure and dynamics are neglected. This constraint is based solely on the limitations of the force prediction model that is currently used. A more sophisticated model that calculates deflections can easily be incorporated when available. The location of the cutting tool can be adjusted accordingly to account for this during the engagement calculations.
- *2D modeling of workpiece geometry*: In turning operations, as the workpiece rotates, the tool moves longitudinally along the rotational axis, hence the tool actually sweeps out a 3D helical volume. Since as discussed in the previous section, the cutting forces are related to the uncut chip area, little accuracy is lost in reducing the problem to manipulating 2D cross sections of the workpiece and swept volume. Again, this assumes that dynamics are not considered. If this were not the case then the true impact of vibrations and chatter from process instability on form and surface finish can only be accurately modeled in 3D. A consequence of this simplification is that area calculations need only be made at feed step intervals, i.e., the distance moved per revolution of the workpiece as illustrated in Figure 3.4. While static deflections can be modeled in 2D this is not done in this research.

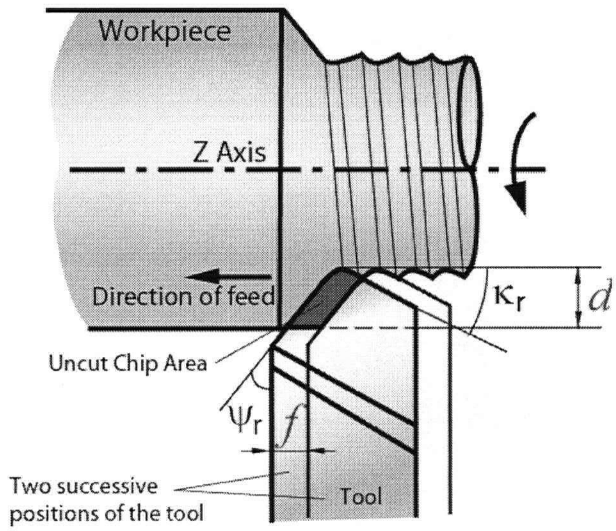


Figure 3.4: 2D Cross Section of Turning Process Showing Feed Step Uncut Chip Area

- *2D modeling of tool geometry*: For the tool workpiece intersection calculations, the region of interest on the cutting tool that defines the uncut chip areas is in the region of the tool nose. As can be seen from Figure 3.5, for a wide range of different tool geometries, this defaults to either a circular edge or two straight edges with an interconnecting circular edge. As is also shown in the figure, these three pieces of geometry can be defined by the tool nose radius (r_ε), the side cutting edge angle (ψ_r), and the end cutting edge angle (κ_r).

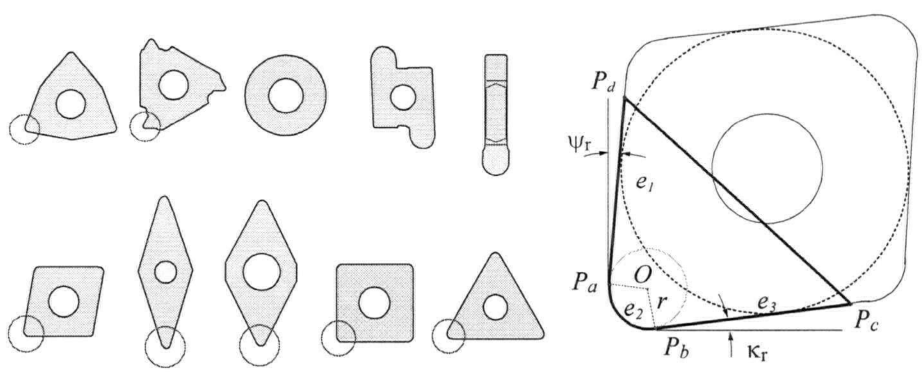


Figure 3.5: Typical Cutting Tool Inserts and Generic Cutting Edge Geometry

Therefore, the tool is constructed by using its major and minor straight cutting edges, nose curve, side cutting edge angle, and end cutting edge angle. Figure 3.6 illustrates the geometric construction of the generic tool shape that is used.

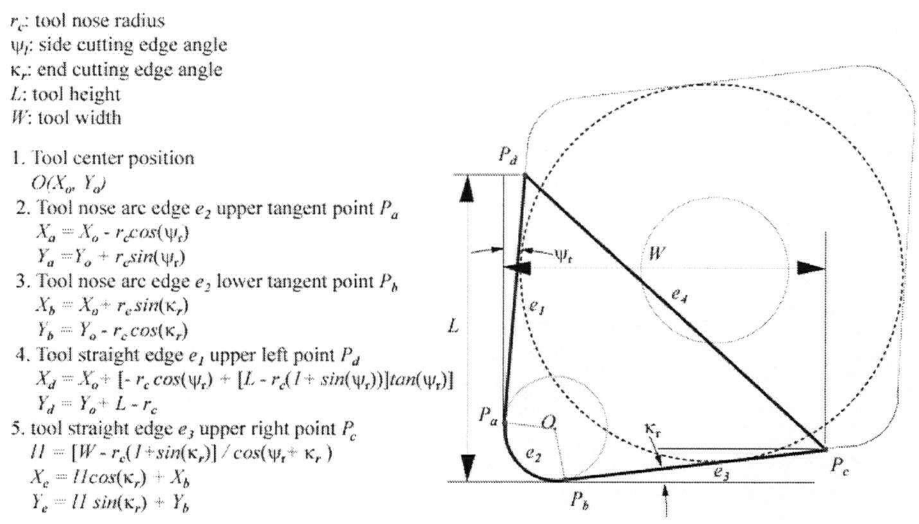
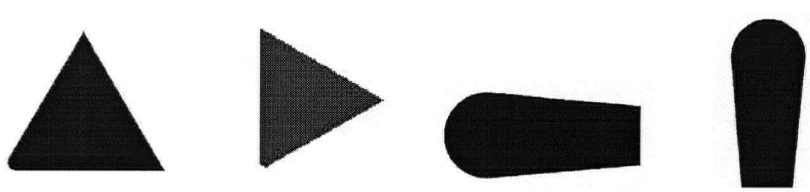


Figure 3.6: Tool Geometry Constructions

This construction method provides sufficient tool geometry information for intersection calculations. Further, this tool model is general enough to be applied to the different types of insert geometries, such as facing tools, contour turning tools, grooving tools, and boring tools as described in case (a), (b), (c), and (d) as shown in Figure 3.7.



(a) Facing (b) Contour Turing (c) Grooving (d) Boring

Figure 3.7: Examples of the Constructed Tools in Virtual Turning System

Since the rake angle of the cutting tool face where the uncut chip area is calculated typically does not lie in the plane of the workpiece cross-section, the face geometry is projected onto this plane. Straight lines project to lines, while the circular edge defining the tool nose radius projects to an ellipse. It is assumed for rake angles typically encountered in practice that this ellipse can be reasonably approximated with a circle with radius equal to the tool nose radius.

- *Tool path geometry:* The toolpaths in this research are followed by the nose centre of the cutting tool. Toolpaths consist of linear and circular components only. Spline toolpaths are assumed to be discretized into small linear segments.
- *Force modeling for contour turning*: The adopted mechanistic cutting force model is verified from the cutting tests in the past only on the longitudinal straight cutting. In that condition, the feed is in the direction of the spindle axis and the depth of cut is constant during machining. In this research, the mechanics of contour turning, along with different types of turning operations, is assumed to apply the same cutting principles. Therefore, when the feed has a machining axis component and a radial component (in contour turning or taper turning), the force model is still assumed to be valid. More cutting tests should be done in the further to verify this force modeling approach.
- *Tool feed direction on contour turning*: Tool feed direction varies along the contour toolpaths, the two feed directions at tool successive positions separated by a feed interval are the tangent vectors along the tool path curve. Due to the feed step is considerably small and usually the tool nose radius is significantly smaller than tool path curve, the difference between two feed vectors is fair small. Therefore, in this research the feed at each feed step is in the direction of the toolpath tangent vector of the tool at current feed step.

- *Depth of cut of contour turning*: The depth of cut of contour turning is defined as a distance from the workpiece boundary, which intersects with the cutting edges of the tool at current step, to the instantaneous feed vector at the tool-part contact point, which is the offset of the feed vector from the tool nose center to the tool curve edge.

3.6 Summary

In this chapter, the Virtual Turning System is outlined. This system is capable of capturing tool-workpiece intersections along the whole toolpaths, and predicting cutting forces, torque and power for the majority of common turning process, such as regular turning, facing, grooving, and boring operations. The system is based on solid modeling technology, and other techniques and algorithms, such as analytical and feature-based methodology, tool construction method, process orientation methodology, are used for speeding up the computation and augmenting the capability of the system. Several basic assumptions of this system are presented for future improvement. In the following chapters the details of the intersection methodology and force prediction in contour turning will be presented.

Chapter 4

A Hybrid Analytical, Solid Modeler and Feature-Based Methodology for Extracting Tool-Workpiece Engagements in Turning

4.1 Introduction

As described in the previous chapter, in the Virtual Turning system, the simulation of turning processes is used to optimize cutting conditions so as to minimize machining cycle time while facilitating production of correctly machined parts from the very first component. However, this process requires an accurate calculation of *Tool-Workpiece Engagement* (*TWE*) geometry to give chip area characteristics used in predicting instantaneous cutting forces, power, and torque at positions along the tool path. This becomes challenging when the initial workpiece geometry has a shape history (e.g., castings or forgings), when the tool path is complex, or when the tool edge is complex, such as during contour turning with formed tools and groove turning.

Solid modelers are increasingly becoming an option for performing these calculations due to the increased robustness and efficiency that is evolving in this technology. These modelers are used to perform Boolean intersections between 2D representations of the cutting tool and the in-process workpiece to extract the engagement geometry. For complicated turned components, particularly those machined from non-cylindrical workpieces, these intersections must be performed at feed increments corresponding to each rotation of the workpiece to guarantee that changes are properly identified. This requirement can easily lead to several thousand Boolean intersections that must be performed to simulate a part. These Boolean operations greatly increase simulation time and the likelihood of modeler errors when intersections between marginal geometry are attempted. Thus, there is a motivation to integrate intelligence into the *TWE* calculations to increase efficiency and improve robustness. This chapter describes research that combines analytical and feature-based methodologies to augment the use of a solid modeler.

Exploiting the cutting tool insert similarities, where they engage the workpiece, and the 2D turning process simplifications make it possible to identify a limited number of engagement conditions that occur over significant regions of each tool pass. In this research these regions are formalized as *In-Cut Material Removal Features* (*mrF*). The use of the term "in-cut" to define these features is to emphasize that they are regions of interest in the in-process workpiece during material removal as opposed to the traditional definition of features that refers to the geometry of the final part. One consequence of the use of these features is the motivation to develop methodologies for extraction and parametrization. Further, within each region it is possible to characterize the engagement geometry at each feed step of the tool (i.e., at each workpiece revolution) as a small set of *Tool Engagement Features* (*teF*). The parameters of each type of *teF* derive from the machining process parameters. These can be combined into the appropriate formulations and solved analytically using Green's Theorem to find uncut chip area characteristics that are used in modeling the cutting forces. While Green's Theorem is not new and is used to find general 2D areas in the solid modeler, the need in these cases for a generic solution necessitates that the calculations be performed numerically. Due to the limited number of *teF* types that have been identified, these can be directly formulated and solved without numerics to reduce computations.

The rest of this chapter is organized as follows. Since a pure solid modeler-based methodology is also an option for engagement, the development of such an approach is described in Section 4.2. This approach serves as a basis with which to extract the *TWE*s and compare the efficiency of the hybrid methodology. As part of this, details about the construction of tool swept areas are given in Section 4.3. Following this methodology, an analytical approach (based on the *teF* classification given above and Green's Theorem) for extracting tool workpiece intersection parameters are presented in section 4.4. Section 4.5 develops the method for extracting material removal features. This is essentially an area decomposition procedure that divides the material removal area into three feature types. Section 4.6 outlines the overall hybrid analytical, solid modeler, and Feature based methodology, which is the combination of the methods described in above sections. A discussion of the implementation details along with results from validation on an industrial aerospace component follows in Section 4.7. The chapter ends with a general discussion of this approach and some directions for future tool-workpiece engagement research.

4.2 Full Solid Modeler-Based Methodology

As described in the previous section, *TWE* is a key issue for modeling the turning process. A Solid Modeler-based methodology is one of the solutions for this problem. The prototype of this methodology originally developed in author's laboratory is shown in Figure 4.1. After being significantly improved for computational efficiency with added functionality for different types of turning in this research, a complete Solid Modeler-based solution is shown in Figure 4.2, which serves as a basis for *TWE*s extraction and is compared to the hybrid solution later.

The ACIS[1] solid modeling kernel is used to modeling and capture the geometry of the workpiece, its in-process state, toolpaths, and the cutting tool, and to perform Boolean operations and other geometric operations in extracting the *TWEs* and their parameters.

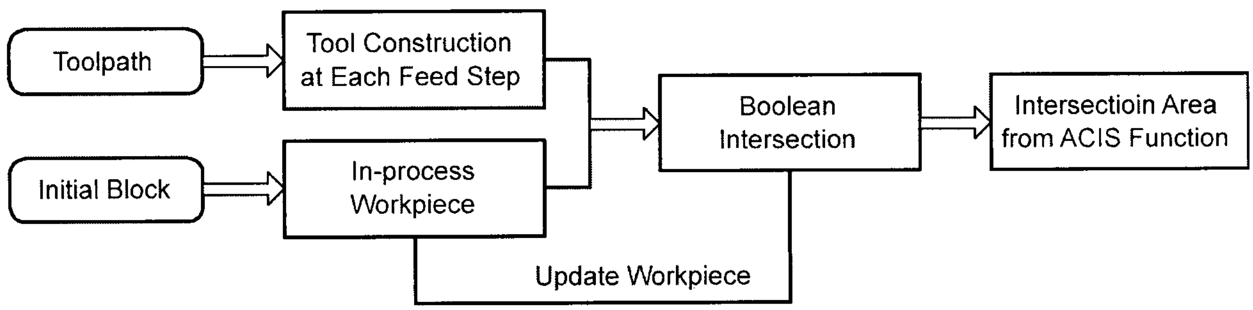


Figure 4.1: Original Solid Modeler-Based Intersection Prototype

Figure 4.1 presents the basic procedures of this methodology. The input requirements are mostly as discussed in Chapter 3. However, the initial workpiece is created as a 2D block by using the bounding box of the workpiece, later the workpiece geometry is extended to the arbitrary model created in the CAD environment in the improved full Solid Modeler solution, which will be described in the following paragraph. Toolpaths are provided as cutter location data generated from a CAM application. In addition to the path geometry, process parameters such as spindle speeds (rpm) or surface cutting speeds (m/min) and feeds (mm/rev) are also contained in this data. Tool geometry is constructed by constant depth of cut, side and end cutting edge angle and tool nose radius. The toolpath is discretized based on the feedrate. The smallest step that can be taken is the feed per revolution (referred to as a feed step). Once the positions along a toolpath have been evaluated, an intersection between the tool geometry and

[1] In this research the ACIS solid modeler product is used to model and manipulate geometry

the in-process workpiece geometry is performed by using the intersection operator in the ACIS kernel, and the intersection area is obtained using ACIS kernel functions. The in-process workpiece is updated by using the Boolean subtraction operator for the intersection at the next feed step.

The main problem of this methodology is that the geometry of the solid in-process workpiece model becomes increasingly complicated as Boolean subtractions remove the tool shape at each feed step. Much of the topological and geometric information stored in the solid model does not contribute to the intersection at a given step. This makes the localization effort for the Boolean operations between the tool and the in-process workpiece solid time consuming and inefficient. To solve this problem, the *Tool Swept Area* (*TSA*), which is the swept region of a tool along a toolpath, is constructed and intersected with the in-process workpiece. Since there is an order of magnitude less toolpaths than total feed steps, the in-process workpiece is less complex than in the original method. Consequently the localization effort is reduced for the Boolean operations and the computational time is improved. Other improvements are also applied to increase the efficiency further for the complex toolpath and workpiece. Details are given in the full Solid Modeler methodology presented in the following paragraphs.

Figure 4.2 (shown in the next page) gives an overview of the full Solid Modeler-based *TWE* extraction methodology. The inputs required are the toolpaths, the initial workpiece geometry, and tool geometry information. In the first step of this solution, the initial workpiece is represented as a 3D ACIS solid model obtained either directly or through STEP translation from the CAD system where the model was originally created. A 2D cross section of the initial workpiece for the *TWE* calculations is obtained by slicing the model with a plane through the machining axis. For convenience and consistency with the axis configuration on lathes, the XZ plane is used, and the tool is projected onto the same plane.

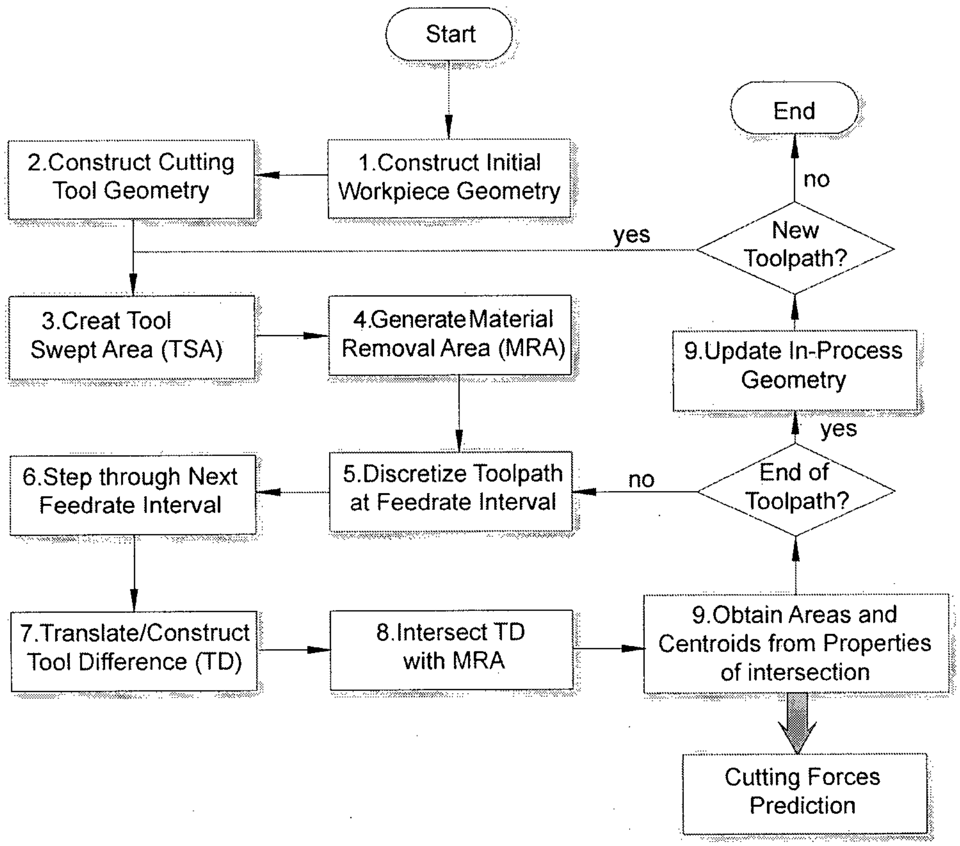


Figure 4.2: Full Solid Modeler-Based Turning Simulation Methodology

As described in Chapter 3, the cutting tool is originally constructed in step 2 by making an edge loop composed of a side cutting edge, an end cutting edge, and a tool nose curve, then converting to a solid model. When the tool does facing, grooving, and boring, a machining set-up angle and a machining direction are given to transform the tool to the correct orientation with respect to the specified machining operation.

The *TSA* is constructed in step 3 based on the tool geometry and the toolpath. A swept area is the union of the area occupied by the tool at all positions during motion over that path. The *TSA* is generated by constructing its boundary (consisting of line and arc segments) as a sequence of edges to which a face is added by the modeler to give a closed 2D shape. Since the *TSA* is very important for correct workpiece updating and *TWE* calculation, its construction algorithm will be described in detail in the next section.

The *Material Removal Areas* (*MRA*) can be obtained by performing a Boolean intersection between the *TSA* and the in-process workpiece (step 4). In Step 5 the toolpath is discretized based on the feedrate. To speed up computation times, intersections at only a few steps need to be calculated when the engagement is not changing. Determining when these invariant engagement conditions occur is part of this research in another solution.

The *TWE* of each feed step can be viewed as the Boolean difference between the tools at successive positions separated by the feed step intersected with the *MRA*. Therefore, two consecutive positions of the tool along the toolpath need be identified in Step 6, and the Boolean difference between these two tools (*TD*) is generated in Step 7. For linear toolpaths, the *TD* is constant and only a transformation is needed to locate it for different feed steps. But for a curved toolpath, *TD* needs to be generated at each feed step. Step 8 performs the engagement calculation by intersecting the *TD* with the *MRA*. Compared to the original prototype shown in Figure 4.1, the intersection between the *TD* and *MRA* at each feed step is another important improvement made in this research. This is because the complexity of the *MRA* does not increase continuously as the tool goes through more toolpaths.

To match the input requirements of the force prediction model adopted in this research, the *TWE* is decomposed into sub-regions due to the difference in cutting mechanics over these regions. Areas and centers of gravity for each of these regions are then calculated by extracting face properties using ACIS kernel functions.

The tool position is incremented (Step 6), and the calculations performed in Steps 6, 7, 8, and 9 are repeated until the chip areas at all positions for a given tool path have been evaluated. Step 3 is repeated until all toolpaths have been processed. Areas and centers of gravity are saved to a file that is subsequently used by the force prediction model. This approach is valid only under the assumption that the workpiece, tool, and surrounding machine tool structure are rigid. If compensation of the tool location due to flexure and dynamics are to be considered then the process model needs to be evaluated at each position of the cutting tool and deflection information feedback to adjust the location of the tool at the next step.

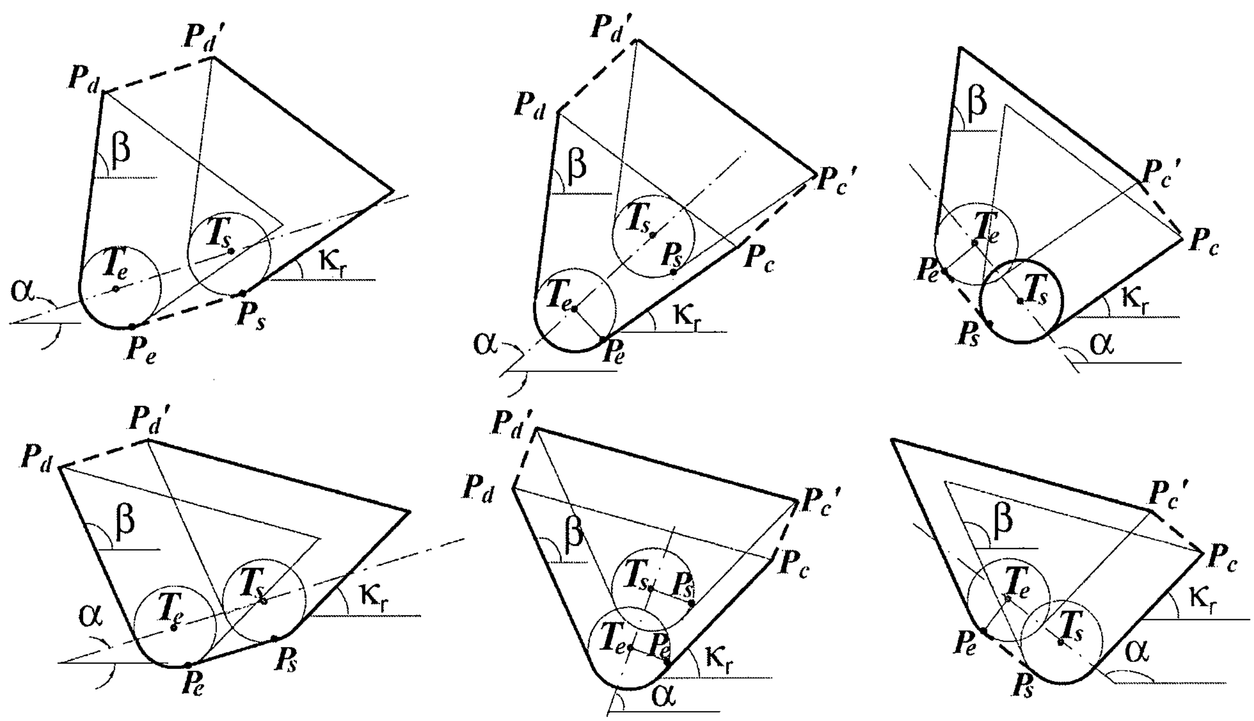
The advantages of this methodology are that it is simple to implement, and it is a generic solution that works regardless of the complexity of the workpiece geometry. The disadvantages are that performing Boolean operations at each step are computationally expensive, and surface-surface intersections in solid modelers are not fully stable, particularly when marginal overlaps between the tool and workpiece occur. Boolean operations may fail in these cases. By reducing the number of operations that need to be performed, computational efficiency can be increased and the likelihood of modeling errors reduced (though not completely eliminated). Towards this goal, analytical area and centroidal calculations and the use of features will be described in Section 4.4 and 4.5.

4.3 Tool Swept Area (*TSA*) Construction

The *TSA*, analogous to the Swept Volume in 3D is the total area that a tool occupies over one toolpath. It is constructed by a series of ordered edges, which include tool boundary edges and new envelope edges. The construction of the *TSA* for each toolpath requires finding the outer or/and inner envelope points of these edges, then connecting all together to form an edge loop. Using ACIS functions to convert the edge loop to a solid body, the *TSA* is represented as a solid model for Boolean operations to be performed with the in-process workpiece solid model. For different types of toolpaths, i.e., linear toolpaths and circular toolpaths, the *TSA* construction methods are different. These are described in section 4.3.1 and section 4.3.2.

4.3.1 Linear Toolpath *TSA* Construction

Linear toolpath *TSA* has three cases with respect to edge loop connection as shown in Figure 4.3. ψ_r is the side cutting edge angle, κ_r is the end cutting edge angle, α is the toolpath angle, and β is the complementary angle of ψ_r, $\beta = \pi/2 - \psi_r$. T_s, T_e are the start and end positions of the toolpath, P_sP_e is the common tangent line offset from T_sT_e, and P_s, P_e are the corresponding tangent points of the tool at T_s, T_e positions. P_d, P_d' are the side cutting edge upper points of the tool at T_e, T_s positions (refer to Tool Geometry Construction as shown in Figure 3.6). Similarly, P_c, P_c' are the end cutting edge upper points of the tool at T_e, T_s positions.



case 1	case 2	case 3
$\alpha < \beta, \kappa_r$	$\kappa_r < \alpha < \beta$	$\alpha > \beta, \kappa_r$

—— Tool boundary edge —·— Tool path - - New envelope edge

Figure 4.3: Three Cases of Tool Swept Area of Linear Toolpath

In case 1, when $\alpha < \beta, \kappa_r$, two new envelope edges are formed by P_sP_e and P_dP_d', other *TSA* edges are constructed by tool edges at T_s and T_e positions. In case 2, when $\kappa_r < \alpha < \beta$, new envelope edges are P_dP_d' and P_cP_c'. Finally, in case 3, when $\alpha > \beta$ and κ_r, new envelope edges are P_sP_e and P_cP_c'. Since α, β and κ_r are known before the construction, it is straightforward to classify a linear toolpath *TSA* into one of the three cases. In each case, the *TSA* is constructed by connecting the corresponding new envelope edges and tool edges that have been identified. This method is specific to modeling swept areas for turning operations based on the generic tool geometry of figure 3.6. However, a general linear toolpath *TSA* algorithm is desirable for all there cases, and it is in fact simple and straightforward as described in the follows.

It can be seen that a *TSA* of a linear toolpath is the convex hull enclosing the tool shape at the start and end toolpath positions. To achieve this, the upper furthest point and lower furthest point of the tool, with respect to the toolpath, need to be identified and connected to form the new envelope edges, as illustrated in Figure 4.4 . The procedure is presented in the following algorithm.

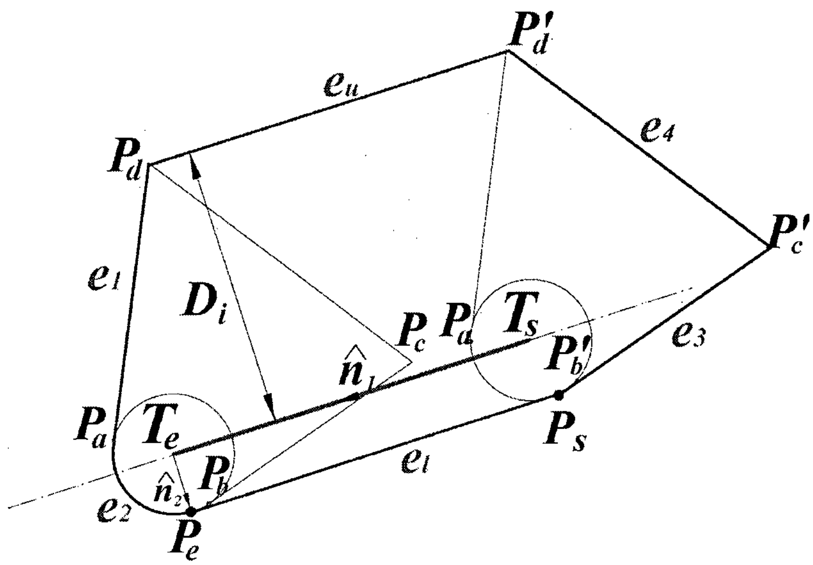


Figure 4.4: Linear Toolpath Tool Swept Area Construction

Algorithm **Linear Toolpath *TSA* Construction**

INPUT: $Tool_i$ (ψ_r, κ_r, r_ε, P_a, P_b, P_c, P_d), $i = 1$ *to* 2, toolpath T_sT_e

OUTPUT: *TSA* boundary edge loop

STEP:

1. P_s, P_e $\leftarrow$ toolpath T_sT_e offset r_ε distance

2. D_i, ($i = a$ *to* e) $\leftarrow$ signed distance between P_i ($i = a$ *to* e) to T_sT_e.

3. Find Upper furthest point P_u

   where $D_u < 0$ and $D_u = max(|D_i < 0|)$

4. Find Lower furthest point P_l

   where $D_l > 0$ and $D_l = max(|D_i > 0|)$

5. New upper envelope edge e_u

   $\leftarrow$connect P_u of the tool at toolpath start and end positions

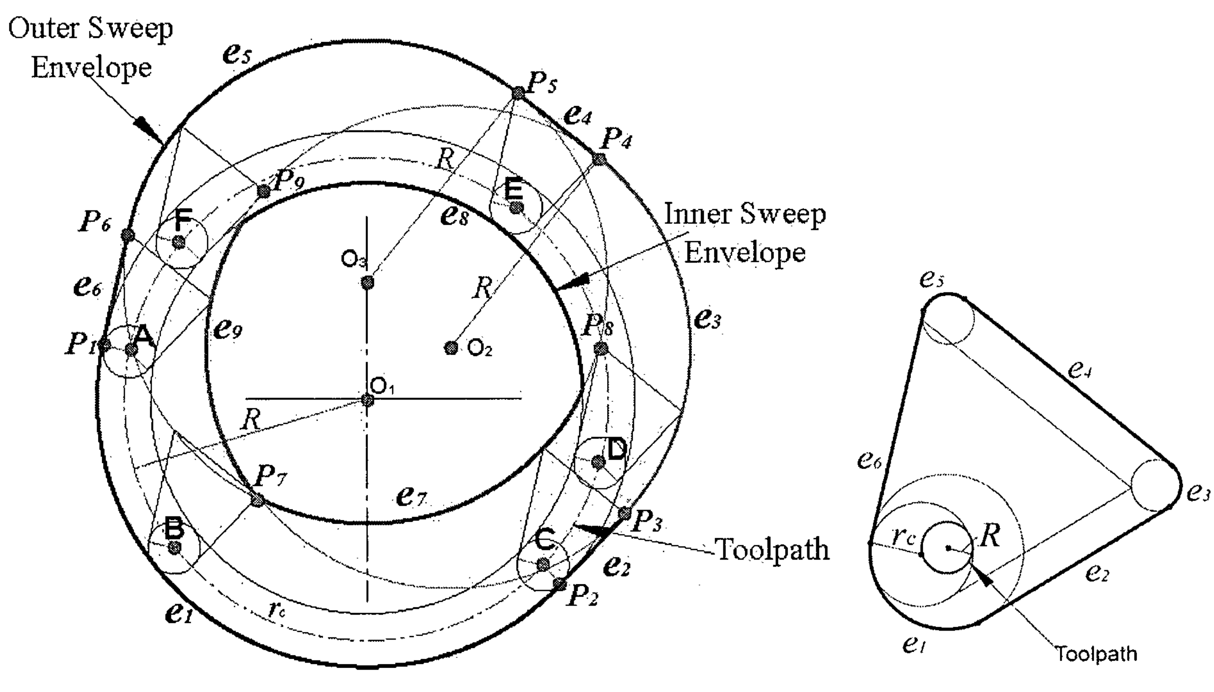
6. New lower envelope edge e_l

   $\leftarrow$ connect P_l of the tool at toolpath start and end positions

7. *TSA* boundary edges $\leftarrow\{\{e_e\}_{PuPl}, e_l, \{e_s\}_{PlPu}, e_u\}$

In step 1, the offset tangent point $P_e = T_e + r_\varepsilon \cdot \hat{n}_2$, where $\hat{n}_2$ is the unit vector perpendicular to toolpath unit vector $\hat{n}_1$ (P_s is obtained similar to P_e). Let L denote the toolpath length $|T_eT_s|$, $\hat{n}_1 = \dfrac{T_e - T_s}{|T_eT_s|} = \dfrac{T_e - T_s}{L}$, $\hat{n}_2 = (-n_{1y}, n_{1x})$. Step 2 finds the distance D_i from each tool point P_i (i from a to e) to the toolpath T_eT_s. $D_i = |(P_i - T_e) \times \hat{n}_1|$. For a 2D case, this equation reduces to $D_i = \dfrac{1}{L}[(x_i - x_s)(y_e - y_s) - (y_i - y_s)(x_e - x_s)]$, where $P_i(x_i, y_i)$, $T_e(x_e, y_e)$, and $T_s(x_s, y_s)$ give the x and y coordinate notations. It must be noted that, D_i is a signed distance. In the upper region with respect to the toolpath, the distance D_i is negative, and the furthest point P_u corresponds to the largest negative D_i in Step 3. Similarly, the distance D_i is positive in the lower region, and the furthest position P_l corresponds to the maximum positive D_i in Step 4. These furthest points at the start tool position and end tool position are connected to form the new envelope edges. In the last step, the *TSA* boundary edges are composted of three parts, two envelope edges e_u and e_l, tool boundary edges $\{e_e\}_{P_uP_l}$ from P_u to P_l at the end toolpath position, and the tool boundary edges $\{e_s\}_{P_l P_u}$ from P_l to P_u at the start toolpath position. In the example shown in Figure 4.4, the *TSA* is $\{e_l e_2 e_l e_3 e_4 e_u\}$.

4.3.2 Circular Toolpath *TSA* Construction



(a) $R > r_\varepsilon$ (b) $R < r_\varepsilon$

Figure 4.5: Circular Toolpath Tool Swept Area Construction

The above figure illustrates the swept area generated by the tool sweeping along a circular path. In Case (a) the toolpath radius R is bigger than the tool nose radius r_ε. In Case (b) R is smaller than r_ε. The difference between these two cases is that there is no inner sweep envelope in the *TSA* of (b) since the entire interior area is swept out. The Swept Area of (a) is the region between two edge rings, the Outer Sweep Envelope, and the Inner Sweep Envelope. The Outer

Sweep Envelope consists of three circular edges e_1, e_3, e_5 and three straight edges e_2, e_4, e_6. The first three edges are generated from sweeps of the tool nose curve, the tool upper right point P_c, and the tool upper left point P_d (See Figure 3.6). The latter three edges come from the tool boundary edges that are tangent to the first three circular edges respectively. Similarly, the Inner Sweep Envelope consists of three circular edges e_7, e_8, and e_9, which are generated from the sweeps of the tool upper left point P_d, the tool nose curve, and the tool upper right point P_c respectively.

To construct the *TSA* of the circular toolpath, the outer and inner sweep envelope edges need to be generated. The corresponding critical points P_1 to P_9 need to be calculated. Note that P_8 is a transient position, below which the inner envelope edge e_7 is formed by P_d of the tool. This is because the angle of the tangent vector of the sweeping envelope formed by P_d at P_8 is just the tool side cutting edge angle (ψ_r). When the angle of the tangent vector is bigger than ψ_r (below), the tool motion is covered by this sweeping envelope e_7. Similarly, P_9 is the critical position below which the inner envelope edge e_9 is formed. The angle of the tangent vector of the sweeping envelop formed by P_c below P_9 is bigger than the tool end cutting edge angle (κ_r). According to the properties of common tangents and planar rigid motion, these points can be obtained easily. For example, P_1 and P_2 can be formulated as follows:

$$P_1: \quad \begin{aligned} X_1 &= X_{O1} - (R + r_\varepsilon)\cdot\cos(\psi_r) \\ Y_1 &= Y_{O1} + (R + r_\varepsilon)\cdot\sin(\psi_r) \end{aligned} \tag{4.1}$$

$$P_2: \quad \begin{aligned} X_2 &= X_{O1} + (R + r_c)\cdot\sin(\kappa_r) \\ Y_2 &= Y_{O1} - (R + r_c)\cdot\cos(\kappa_r) \end{aligned} \tag{4.2}$$

Where $O_1\,(X_{O1},\ Y_{O1})$ is the center of the toolpath. Other position calculations are presented in Appendix A.

According to these positions and the tool geometry relationship, the corresponding critical toolpath positions A to F can be identified. Six sections of toolpath are constructed, i.e., AB, BC, CD, DE, EF and FA. In real turning operations, it is common that only a portion of a circular tool path be encountered. To determine the tool swept area of a given toolpath it is first necessary to find which section or which combined sections the path belongs to. Then the tool swept area is constructed by generating the outer and inner edge loops based on the boundary conditions of those sections. Details of this circular toolpath *TSA* construction are presented in Appendix A.

4.4 Green's Theorem-based Analytical Intersection Area Calculation

As briefly discussed in the introduction, the disadvantages of the full Solid Modeler methodology are that performing Boolean operations at each step is time consuming and that surface-surface intersections in solid modelers may fail in particular cases. Also the generic nature of area calculations performed by a solid modeller requires a general-purpose numerical solver. As computational speed is a critical concern, efficiency can be increased by applying an analytical solution directly to calculate the uncut chip area. This improvement means that Boolean operations need not be preformed at each step to obtain the chip shape, if the boundary information is provided. A Feature recognition methodology can first be used to identify the boundary conditions as described in next section.

It is possible to characterize the engagement geometry as a small set of *Tool Engagement Features* (*teF*). The parameters of each type of *teF*, such as area, chip-cutting edge contact length, and gravity center, derive from the machining process parameters, i.e., these *teF* parameters can be expressed as functions of feedrate, cutting speed and tool geometry. These *teF* expressions can be combined into several appropriate formulations and solved analytically using *Green's Theorem* to find the uncut chip area characteristics that are used in modeling the cutting forces. The formulations of this solution using *Green's Theorem* will be the subject of this section.

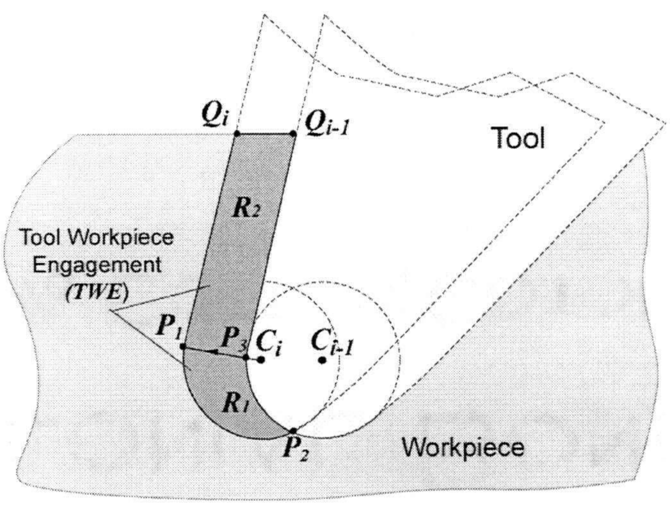


Figure 4.6: Tool Workpiece Engagement (*TWE*)

Green's Theorem is widely applied in the study of mathematics; it can convert the double integral to a line integral over its boundary.

$$\iint_R \left(\frac{\partial f}{\partial x} + \frac{\partial g}{\partial y}\right) dxdy = \oint_C (fdy - gdx) \tag{4.3}$$

The area and centroid calculation are two of the main applications of Green's Theorem. Figure 4.6 illustrates the *Tool Workpiece Engagement* (*TWE*). The boundary edges consist only of lines and arcs. After the equations of lines and arcs are specified as parametric equations $(x(t), y(t))$, area equations are expressed as follows. The three formats are equivalent. In this research the middle format is used in *TWE* calculations for better geometric understanding.

$$A = \int_1^2 xy'\,dt = \int_1^2 -yx'\,dt = \frac{1}{2}\int_1^2 (xy' - yx')dt \tag{4.4}$$

Mathematically, the *TWE* can be expressed as follows:

$$TWE = \oint e_i(P_i, P_{i+1}, P_i(u), r_\varepsilon, C_i) \tag{4.5}$$

where the boundary is a set of connected edges of arc or line type:

$$P_i(u) = \begin{cases} (1-u)P_i + uP_{i+1} & 0 \le u \le 1 \quad \textit{Line segment} \\ C_i + \begin{bmatrix} r_c\cos(u) & r_c\sin(u) \end{bmatrix} & \theta_i \le u \le \theta_{i+1} \quad \textit{Arc} \end{cases} \tag{4.6}$$

and

e_i: Edge on the boundary of *TWE*

P_i, P_{i+1}: End points of edge e_i

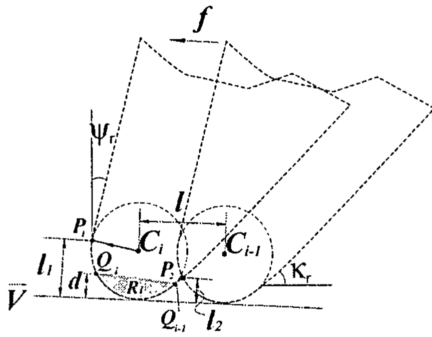
C_i: Center of tool nose along the tool path

r_ε: Tool nose radius

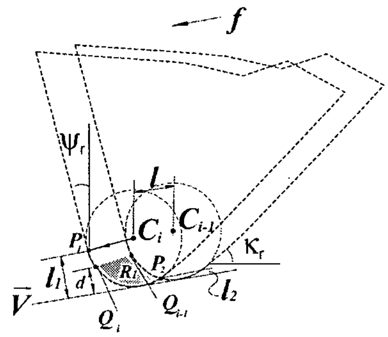
θ_i: Angle of vector P_iC_i in world coordinate system.

In general, two or more edges define the boundary of a *teF*. These edges are either portions of the in-process workpiece boundary before the machining of the current material removal area (*MRA*) or portions of the cutting tool boundary. These groups are referred as the sets $W\{e_i\}$ and $T\{e_i\}$ respectively. This is sufficient for identifying the various combinations of engagement that are possible for any turning part under consideration.

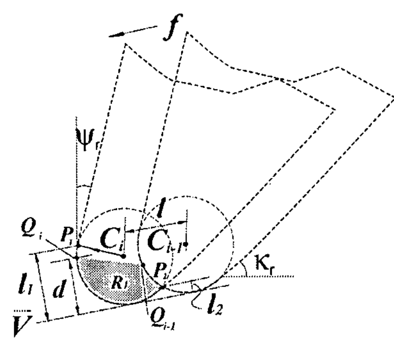
Figure 4.7 illustrates six commonly occurring *teFs* that have been identified for generic turning inserts such as those described in the previous chapter that cover a wide range of machining conditions.



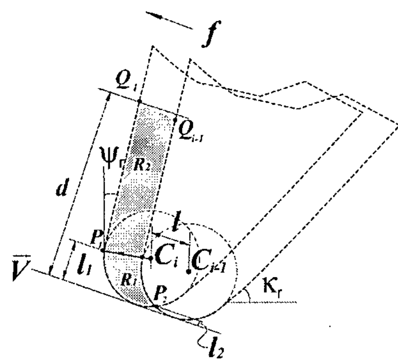
Case 1. $d < l_2$



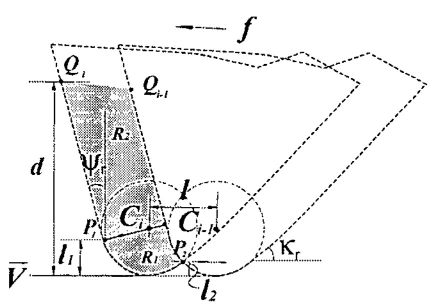
Case 2. $l_2 < d < l_1$. $f < r_\varepsilon$



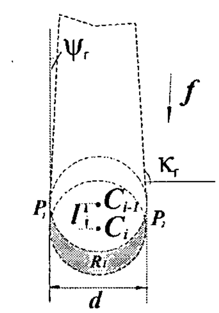
Case 3. $l_2 < d < l_1$. $r_\varepsilon < f < 2r_\varepsilon$



Case 4. $d > l_1$. $f < r_\varepsilon$



Case 5. $d > l_1$. $r_\varepsilon < f < 2r_\varepsilon$



Case 6. Grooving

Figure 4.7: Classes of Generic Tool Engagement Features (*teF*)

Parameter	Constraints	teF1	teF2	teF3	teF4	teF5	teF6
Depth of cut (d) mm	$d < l_2$	√					
	$l_2 < d \leq l_1$		√	√			√
	$d > l_1$				√	√	
Feedrate (mm/rev)	$0 < f \leq r_c$	√	√		√		√
	$r_c < f \leq 2r_c$	√		√		√	
ψ_r (deg)	$\psi_r \geq 0$	√	√	√	√	√	√
	$\psi_r < 0$	√	√	√	√	√	√
Regions of teF	R_1	√	√	√			√
	R_1+R_2				√	√	

Table 4.1: Conditions of Generic Tool Engagement Features (*teF*)

Each *teF* is defined parametrically corresponding to the depth of cut (*d*), the feedrate (*f*), and the insert lead angle (ψ_r). As indicated in the accompanying table (Table 4.1), feature types are differentiated by constraints applied to these parameters. Depth of cut (d), which is the distance from the workpiece boundary to tool contact point measured perpendicular to the instantaneous feed direction, determines the number of sub-regions that makeup the *TWE*. Feedrate (f) indicates how far the tool moves during one revolution of the workpiece. Its direction is an instantaneous tangent vector of the tool feed motion. The tool nose radius r_ε and the side cutting edge angle ψ_r are geometric properties of the turning tool. In Figure 4.7, l is the distance between the successive two tool positions C_iC_{i-1}. If the tool path is a straight line, l is equal to feedrate f. If the tool path is a circular edge, f is the arc length and l is the chord length. P_1 is the tool nose curve upper tangent point of the tool, and P_2 is the intersection point between the two tools. $\vec{V}$ denotes the instantaneous feed direction that is tangent to the tool nose curve at

C_i, and l_1 and l_2 are the distances from P_1, P_2 to $\vec{V}$ respectively. Appropriate analytical equations can be formulated for each case by giving the boundary conditions of the engagement, such as intersecting positions and angles.

For case (1), the type of *teF1*, d is smaller than l_2 ($P_2\vec{V}$), there is only one curve region R_1, the boundary of which is composed of two edges. One is tool nose curve, and the other is workpiece boundary edge. Since d is fairly small, the workpiece boundary edge within one feed interval can be regarded as a straight line. Geometric equations for the uncut chip area A_1, chip-tool contact length Lc_1, and gravity center G_1 of the region are as follows:

$$A_1 = \frac{1}{2} r_\varepsilon^{\ 2} (\phi - \sin(\phi)) \tag{4.7}$$

$$L_{c1} = r_\varepsilon (\phi - \sin(\phi)) \tag{4.8}$$

$$GC_i = \frac{4}{3} r_\varepsilon \left(\sin^3(\phi)\right) \Big/ (\alpha - \sin(\phi)) \tag{4.9}$$

where $\phi = 2\cos^{-1}\left(\dfrac{r_\varepsilon - d}{r_\varepsilon}\right)$.

For *teF2* and *teF3* as shown in case (2) and case (3), d is lies between l_1 and l_2, i.e., only the curve region A_1 lies inside the engagement, while f is smaller than tool nose radius r_ε for *teF2* and bigger in *teF3*. The difference between the two cases is the possible type of P_2 (the intersection point of cutting edges of two tools): it can be nose arc-nose arc intersection in both cases, or nose arc-side cutting edge intersection in *teF2*, but end cutting edge- nose arc intersection in *teF3*. Both cases need to identify the equivalent side cutting edge angle ψ_r' with respect to the instantaneous feed direction at the successive locations, i.e., $\psi_r' = \psi_r + \alpha$, where

α is the toolpath angle. For *teF4* and *teF5* as shown in cases (4) and case (5), two regions R_1 and R_2 are included in the engagement. *teF6* is a special case for straight groove cutting.

To describe all the cases illustrated above, a common area calculation algorithm is expressed in this section, which is based on a general Green's Theorem-based line integral. Details of the developed equations for each case are listed in Appendix B.

Assuming the area integrated by a straight line e_l is A_l, and that integrated by an arc e_a is A_a, the parametric equations of e_l and e_a are,

$$e_l: \quad \begin{cases} x = (1-u)X_i + uX_{i+1} \\ y = (1-u)Y_i + uY_{i+1} \end{cases}, \quad 0 \le u \le 1 \tag{4.10}$$

where the end points are $P_i(X_i, Y_i), P_{i+1}(X_{i+1}, Y_{i+1})$

$$e_a: \quad \begin{cases} x = X_c + r\cos u \\ y = Y_c + r\sin u \end{cases}, \quad \theta_1 \le u \le \theta_2 \tag{4.11}$$

where the centre point is $C(X_c, Y_c)$, θ_1, θ_2 are the parametric bounds for the circular edge.

$$A_l = \int_0^1 -yx'\,du = \frac{(y_{i+1} + y_i)(x_{i+1} - x_i)}{2} \tag{4.12}$$

where $P_{i+1}(x_{i+1}, y_{i+1})$, $P_i(x_i, y_i)$

$$A_a = \int_{\theta_1}^{\theta_2} -yx'\,du = \frac{r^2 u}{2} - rY_c\cos(u) - \frac{r^2}{4}\sin(2u)\Bigg|_{\theta_1}^{\theta_2} \tag{4.13}$$

where θ_1 and θ_2 are the parametric bounds for the circular edge.

Since A_l and A_a are signed areas, the total area A of a region is the sum of all sub-areas integrated by all edges, where *n* is the total number of edges of *A*, *m* is the number of arc edges, and *n-m* is the number of line edges.

$$A = \sum_{i=0}^{n} A_i = \sum_{j=1}^{m} A_{a,j} + \sum_{k=1}^{n-m} A_{l,k} \tag{4.14}$$

Centroidal positions are calculated in a similar manner and will not be discussed in this thesis. A general Green's Theorem-based area calculation algorithm is illustrated with the example (*teF4*) as shown in Figure 4.8.

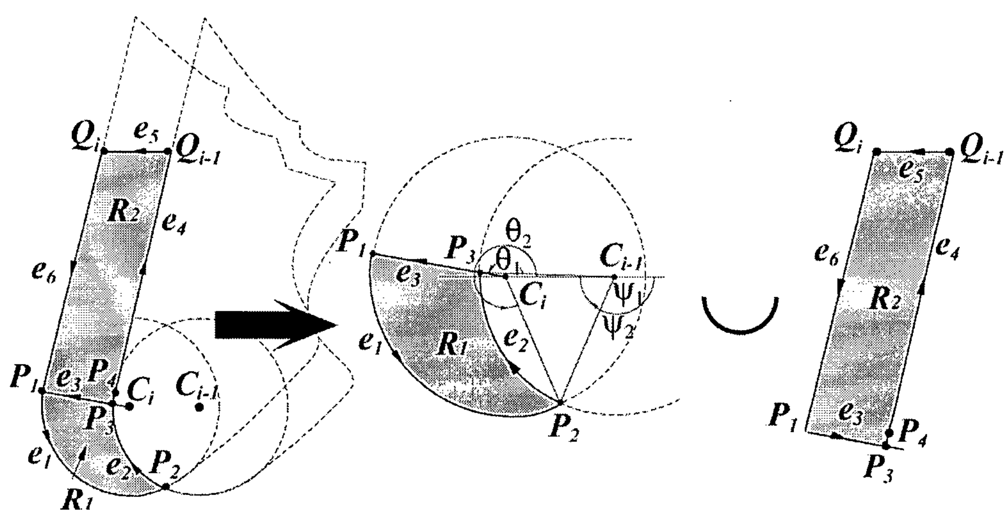


Figure 4.8: One Example of *teF4* Area Calculation

Algorithm **Area_Calculation**

INPUT: $C_i, C_{i-1}, \{P_i\}, e_i, n, m, k, f$

> ($\{P_i\}$: the set of end points, *n*: number of edges, *m*: number of circular edges, *k*: number of zones)

OUTPUT: A_1, A_2, A

STEP:

For region R_j, $j = 1$ to 2

From $i = 1$ to n

CASE geometry_type $(e_i) \equiv$ LINE

$$A_i = \int_0^l - yx'\,du = \frac{(y_{i+1} + y_i)(x_{i+1} - x_i)}{2},$$

where $P_{i+1}(x_{i+1}, y_{i+1})$, $P_i(x_i, y_i)$

CASE geometry_type$(e_i) \equiv$ ARC

If center is C_i

$$A_i = \int_{\theta_1}^{\theta_2} - yx'\,du = \frac{r^2 u}{2} - rY_c\cos(u) - \frac{r^2}{4}\sin(2u)\Big|_{\theta_1}^{\theta_2},$$

where $\theta_1 = \pi + \arctan(y_{P_iC_i} / x_{PiCi})$ $\quad 0 \le \theta_1 < 2\pi$

$\theta_2 = 2\pi + \arctan 2(y_{P_{i+1}C_i}, x_{P_{i+1}C_i})$ $\quad 0 \le \theta_2 < 2\pi$

If center is C_{i-1}

$$A_i = \int_{\psi_1}^{\psi_2} - yx'\,du = \frac{r^2 u}{2} - rY_c\cos(u) - \frac{r^2}{4}\sin(2u)\Big|_{\psi_1}^{\psi_2}$$

where $\psi_1 = 2\pi + \arctan 2(y_{P_iC_i}, x_{P_iC_i})$ $\quad 0 \le \psi_1 < 2\pi$

$\psi_2 = \pi + \arctan(y_{P_iC_i} / x_{PiCi})$ $\quad 0 \le \psi_2 < 2\pi$

$A_j \leftarrow A_i + A_j$

End

End

$A = A_1 + A_2$

End

In Figure 4.8, points C_i, C_{i-1} are the consequence tool nose center positions along the tool path. The interval is $l = |C_iC_{i-1}|$. *TWE* has two regions R_1, R_2 for cutting force calculation. At first, the intersection point P_2 needs to be identified, if the equivalent side cutting edge angle, $\psi_r' = \psi_r + \alpha$, is smaller than a critical angle $\left(\frac{\pi}{2} - \sin^{-1}\left(\frac{1}{2r_\varepsilon}\right)\right)$, P_2 is an intersection point between two curves. If ψ_r' is positive as shown in the above figure, curve region R_1 is known to be bounded by edges $\{e_1, e_2, e_3\}$, and the close to polygonal region R_2 is bounded by edges $\{e_3, e_4, e_5, e_6\}$. e_1, e_2 are a portion of the tool nose curves, and e_3 is the line segment P_1C_i truncated by e_2 at P_3. e_4, e_6 are tool straight cutting edges, and e_5 is a portion of the workpiece boundary edge. The signed areas covered by all the edges are formulated and summed up to give the total area equations. The formulations are given by the closed form equations (4.15) and (4.16). For any *teF* of this class appropriate boundary conditions are applied to these standard equations to get the results.

$$\begin{aligned}A_1 &= \frac{-r_\varepsilon}{4}(4[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)] \\ &+ r_\varepsilon[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)] \\ &- 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1)) + \frac{1}{2}(Y_3 + Y_1)(X_3 - X_1)\end{aligned} \tag{4.15}$$

$$A_2 = \frac{1}{2}\left(x_3y_4 - x_4y_3 + x_4y_5 - x_5y_4 + x_5y_1 - x_1y_5 + x_1y_3 - x_3y_1\right) \tag{4.16}$$

All the boundary conditions (point coordinates and angles) are derivable analytically. P_1, P_4 are the tool nose curve upper tangent points at C_i, C_{i-1}. P_2 is the intersection point between C_i and C_{i-1}. Q_i, Q_{i-1} are the workpiece boundary positions intersected with the tools at the two positions. The coordinates of these boundary points are $P_i(x_i, y_i)$, $Q_i(x_5, y_5)$, and $Q_{i-1}(x_4, y_4)$, and can be calculated using f, d, the tool geometry and the workpiece boundary in the *MRA*. θ_1 and θ_2 are the angles of vector P_1C_i, P_2C_i, and ψ_1, ψ_2 are the angles of vector P_2C_{i-1}, P_3C_{i-1}.

Thus each region can be parametrically defined in terms of area A, chip-cutting edge contact length L_c, the gravity center G, tool side cutting edge angle ψ_r, feedrate f, depth of cut d. A parametric form for any of the *TWEs* shown in Figure 4.6 can be expressed as,

$$teF(R_i(A_i,\ L_{ci}, G_i), f, d, \psi_r, \kappa_r),\ i = 1 \ to\ 2$$

The formulations of the other *teF* cases are listed in Appendix B. A methodology for the identification and expression of the boundary conditions is presented in the following section.

4.5 Feature-Based Methodology

Green's theorem based analytical equations described in the previous section require the boundary positions, i.e., the end points and angles of the edges of *TWEs*. This translates into finding the coordinates of the points P_1, P_2, P_3, P_4, Q_i, and Q_{i-1}, as shown in Figure 4.8. To identify these positions effectively, feature concepts are introduced in this section. A feature identification methodology is developed in this research to achieve this goal.

4.5.1 In-Process Turning Features

A feature-based methodology developed in this research is based on the decomposition of a material removal area (*MRA*) generated during turning into in-process features. Figure 4.9 shows the taxonomy for features generated during turning process, similar to the one proposed by [Yip-Hoi and Huang, 2004] for 2 ½ D milling. Of interest are the in-cut features.

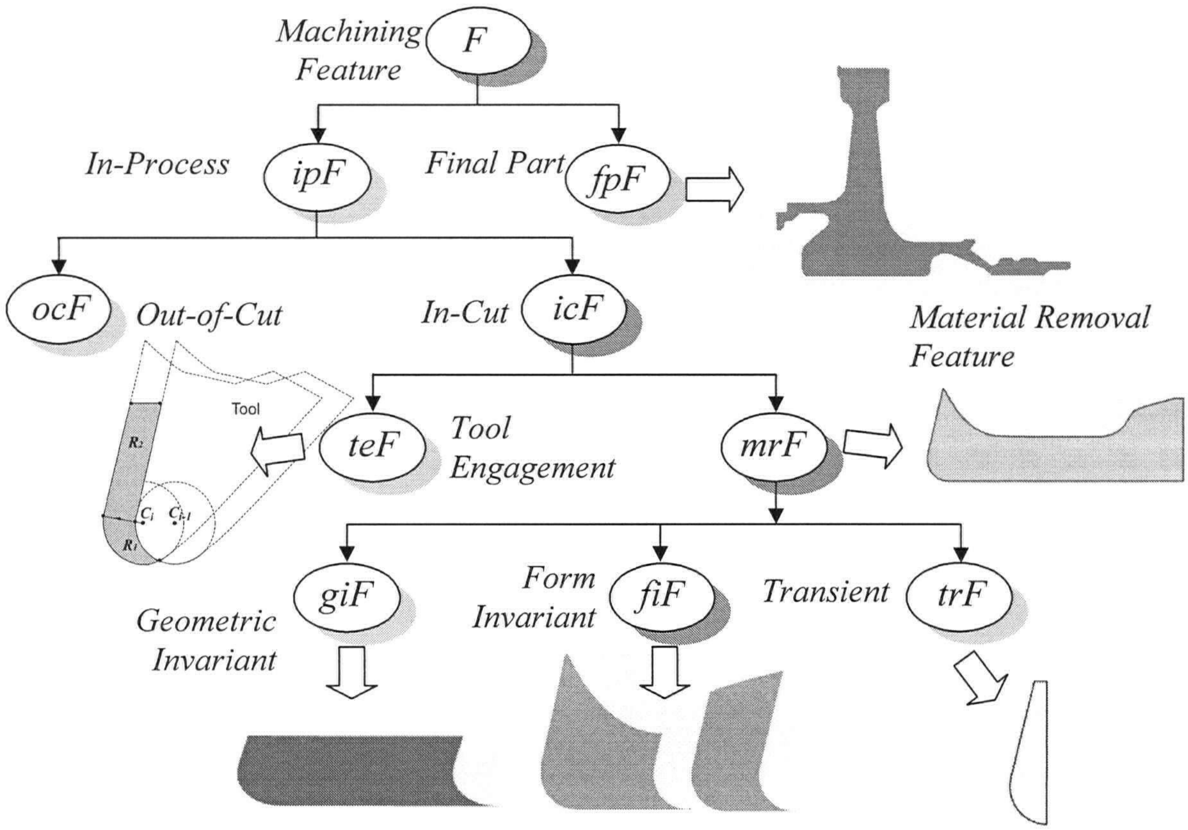


Figure 4.9: Classification of Features Generated from Turning

Tool Engagement Features (*teF*) define the shape of the engagement over a single revolution of the workpiece. Material Removal Features (*mrF*), on the other hand, are regions in the removal volume that correlate with specific types of engagement changes over a complete tool pass (a toolpath). Essentially, for a particular *mrF* the *teF* is of one type. The extraction and parametric expression of a *teF* are fully discussed in the previous section. There are three types of *mrF*. A Geometry Invariant Feature (*giF*) is a region within a tool pass where the geometry of the *teF* at each rotation of the workpiece remains unchanged. This is the case along a linear tool path where the tool cuts at a constant depth of cut. A Form Invariant feature (*fiF*) is one where the class of the *teF*, as defined in Figure 4.7, remains unchanged over the corresponding region.

Hence the shape or topology of the *teF* boundary is fixed, though its geometry varies. In contrast to a *giF*, a *fiF* occurs when the tool is fully engaged with the workpiece and the depth of cut varies continuously over the region defined by the feature. The third type of *mrF* is a Transient Feature (*trF*). These features occur when the tool breaks into or out of the workpiece at the start or end of a pass or when the tool transitions between adjacent regions, as shown in Figure 4.10.

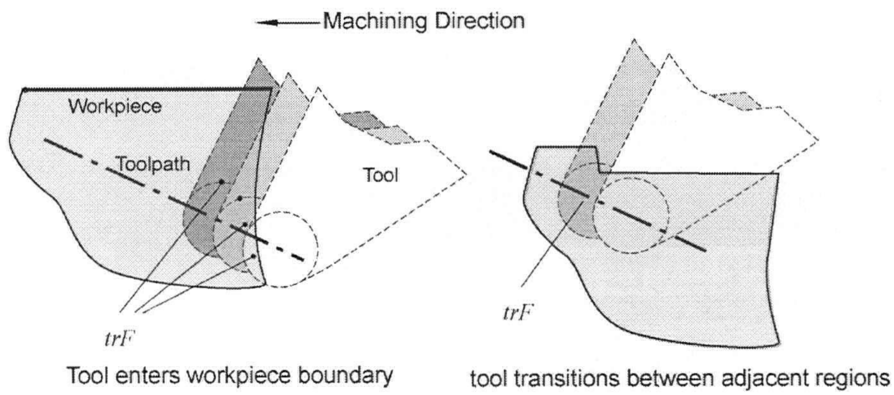


Figure 4.10:Transient Machining Feature (trF)

trFs have unpredictable boundaries and consequently their constituent *teFs* do not fit any of the classes presented in Figure 4.7. The parameters of *trFs* will be extracted by using solid modeler functions, the same as the methodology used in the full Solid Modeler solution. The rest of this section focuses on the parametrization and extraction of *mrFs*.

4.5.2 Extraction of Material Removal Features

Examples of *mrFs* for a single pass are illustrated in Figure 4.11. The *giF* occurs in regions 1, 3, and the *fiFs* correspond to regions 2, 4, and 5. The start region and the regions between *giFs* / *fiFs* belong to the *trF* class. The details of the decomposition one *mrF* into *giFs*, *fiFs*, or *trFs* are presented in this section.

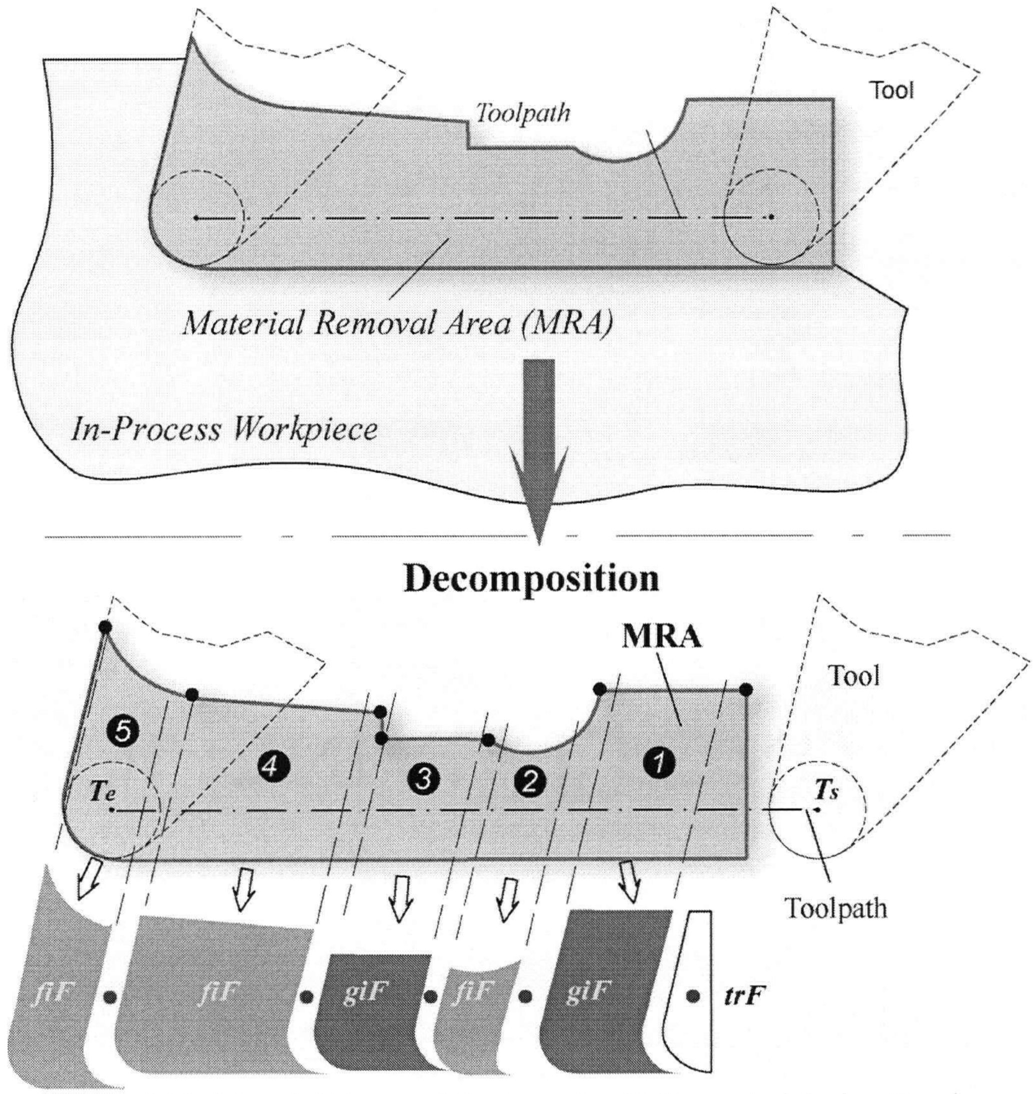


Figure 4.11: Material Removal Features (*mrF*) Generated during Turning

The approach for *MRA* decomposition starts by identifying the edges of the *MRA* boundary that correspond to the workpiece boundary, by traversing *MRA* boundary edges and comparing them with *TSA* and tool boundary edges. The edges that are different from the *TSA* and tool edges are workpiece boundary edges. The start and end positions of these edges are indicated by the points Q_i in Figure 4.12. Figure 4.12 shows a *MRA* which is generated by Boolean intersection between the *TSA* of a circular toolpath T_sT_e and the in-process workpiece.

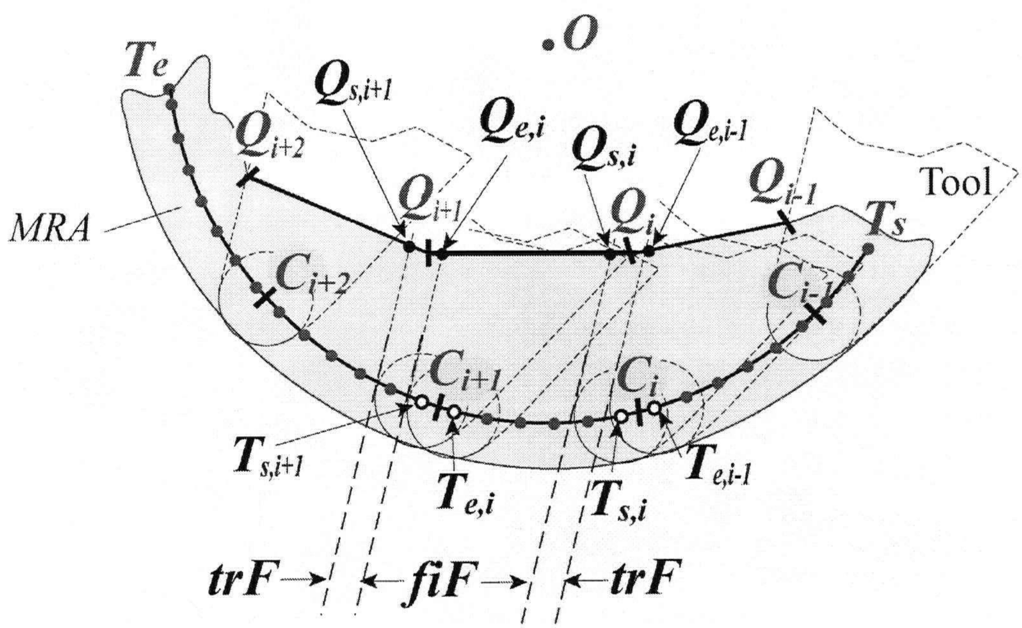


Figure 4.12: *MRA* Decomposition

The toolpath corresponding to the *MRA* is then discretized at the feed step from the start position T_s to its end T_e. These points indicate the tool center positions at each step along this toolpath. These are denoted by the round points T_i in Figure 4.12. Additional strip points C_i are added, corresponding to the tool center positions when the tool leading edge passes though an end point Q_i. If C_i is not coincident with any T_i, this means that between the previous tool location ($T_{e,i-1}$) and the next tool location ($T_{s,i}$), the tool leading edge passes through a discontinuity between two workpiece boundary edges, resulting in a transient engagement feature (*trF*) in this region. *giF* and *fiF* type features occur between adjacent locations of these transition points when the tool leading edge continuously intersects the same workpiece boundary edge. As shown in the example of Figure 4.12, the workpiece boundary edge end point Q_i results in a transition point C_i on the tool path. A *trF* is defined between the feed step points $T_{e,i-1}$ and $T_{s,i}$ that bracket C_i. A *fiF* is present between feed step point $T_{s,i}$ and $T_{e,i}$, which define the start and end tool center positions of the *fiF* on the workpiece boundary edge $Q_{i+1}Q_i$. Identification of feed step points along the tool path as well as the bracketing of different types of engagement features are needed in extracting these features. An algorithm for doing this follows:

Algorithm ***MRA*_Decomposition**

INPUT: *MRA*, Toolpath edge (T_eT_s), *TSA*,

Tool geometry $TG(r_c,\ \psi_r,\ \kappa_r)$, feedrate ($f$).

OUTPUT: *giF* / *fiF*, and *trF*

STEP:

1. $e_1 \ldots e_m$ ← FindWorkpieceBoundaryEdges (*MRA, TSA, Tool*)

2. $T_j = T_s + j \cdot f$ ← DiscritizeToolpathAtFeedrateInterval (T_eT_s, f)

   where $j = 1$ *to* n, n is the number of feed steps, $n = floor(S / f)$, S is the length of toolpath T_eT_s.

3. From $i = 1$ to m (m is the number of workpiece boundary edges)

   for each e_i,

   Q_{i+1}, Q_i ← Get the end points of e_i

   C_{i+1}, C_i ← FindCorrespondingToolPositionfrWB (Q_{i+1}, Qi)

$T_{e,i}$, $T_{s,i}$ ← <u>FindTransitionEngagementPositionsOnToolpath</u>(Q_{i+1}, Qi, C_{i+1}, C_i)

$Q_{e,i}$, $Q_{s,i}$ ← <u>FindWorkpieceBoundaryPositionfromToolPosition</u> ($T_{e,i}$, $T_{s,i}$)

giF_i or fiF_i ← <u>IdentifyMRFforWorkpieceBoundary&Toolpath</u> ($Q_{e,i}Q_{s,i}$ $T_{e,i}$, $T_{s,i}$)

trF_i ← <u>FindtrFBetweenTwoEdges</u> ($T_{s,i}$, $T_{e,i-1}$)

Output giF_i / fiF_i, trF_i

End

To find the tool position from the workpiece boundary or vice versa, the geometric relations between workpiece boundary and tool center are used, and these are also used in *teF* boundary position identification within *giF* / *fiF*, which will be introduced in the following sections and presented in Appendix C in detail.

As shown in Figure 4.12, Q_{i-1} to Q_{i+2} are the workpiece boundary positions, and C_{i-1} to C_{i+2} are the corresponding tool center positions on the toolpath. If the toolpath is a circular line, R denotes the radius of the toolpath, and O is the center of the toolpath. The Transition Engagement Identification algorithm is presented as follows.

Algorithm **FindTransitionEngagementPositionsOnToolpath**
INPUT : C_{i+1}, C_i, T_s, (R, O if toolpath is an arc)
OUTPUT : $T_{e,i}$, $T_{s,i}$
STEP:
1. Get the length between C_j and T_s ($j = i$ and $i+1$)

$$S = \begin{cases} \left|C_jT_s\right| & \textit{Linear Toolpath} \\ \cos^{-1}\left(\dfrac{C_jO \cdot C_jT_s}{R^2}\right)R & \textit{Circulaar Toolpath} \end{cases}$$

2. If C_j corresponds to the start position of workpiece boundary edge,

$$n = ceiling\left(\frac{S}{f}\right),$$

$$T_{s,i} = T_s + n \cdot f$$

If C_j corresponds to the end position of the workpiece boundary edge,

$$n = floor\left(\frac{S}{f}\right),$$

$$T_{e,i} = T_s + n \cdot f$$

End

Note that while this algorithm identifies the transition positions, these are also equivalent to the end positions of *giFs* and *fiFs* along the toolpath. In the following sections, the *giF* and *fiF* identification and parametrization are presented.

4.5.3 Geometric Invariant Machining Feature (*giF*)

To recognize of a *giF* from any non-transient *mrF*, it is easy to see that it is sufficient to determine that the workpiece boundary and toolpath are parallel straight lines. As shown in Figure 4.13, there are two cases for this feature, one is when the workpiece boundary Q_sQ_e intersects the straight cutting edge of the tool, and the other when the intersection occurs on the tool nose curved edge.

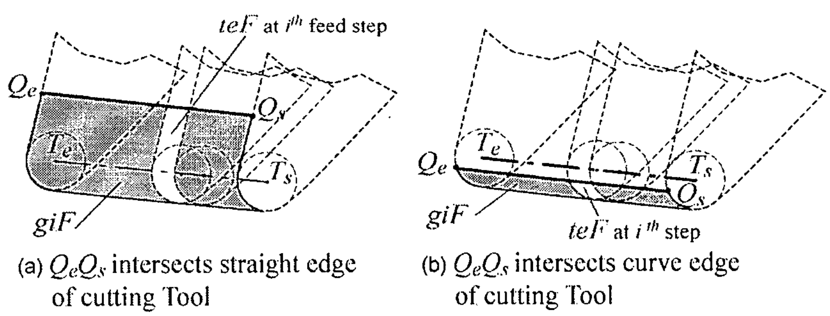


Figure 4.13: Geometry Invariant Features *giF*

Since in a *giF* all *teFs* (at each feed step) have the same geometry, the extraction of the *teF* is performed once for the entire *giF*. Figure 4.14 illustrates an example. With the tool position at C_i and the previous position at C_{i-1}, the corresponding workpiece boundary positions Q_i, Q_{i-1}, the depth of cut (d), and the intersection boundary points P_1, P_2, and P_3 are calculated using 2D linear and circular components intersections. These positions can be expressed as a function of the tool nose center position C_i. Since C_i is on the known toolpath T_sT_e, it can be easily calculated from the number of feed steps.

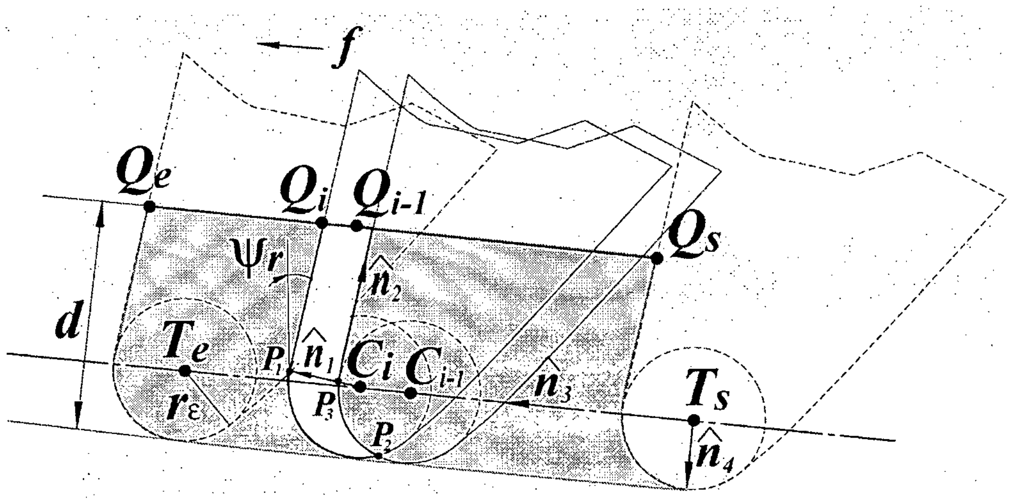


Figure 4.14: *teF* Extraction from *giF*

The boundary points are calculated as follows:

$\hat{n}_1$ is the unit vector of P_1Ci, $\hat{n}_3$ is the unit vector of T_eT_s,. $\hat{n}_2, \hat{n}_4$ are the unit vectors perpendicular to $\hat{n}_1, \hat{n}_3$ respectively. Then,

$$\hat{n}_1 = [-\cos\psi_r, \sin\psi_r] \tag{4.17}$$

$$\hat{n}_2 = [\sin\psi_r, \cos\psi_r] \tag{4.18}$$

$$\hat{n}_3 = \frac{(T_e - T_s)}{|T_e T_s|} \tag{4.19}$$

$$n4 = [-\hat{n}_{3y}, \hat{n}_{3x}] \tag{4.20}$$

$$P_1 = C_i + r_\varepsilon \cdot \hat{n}_1 \tag{4.21}$$

$$P_2 = C_i - \frac{f \cdot \hat{n}^2}{2} + \frac{\sqrt{4r_\varepsilon^{\ 2} - f^2}}{2f} \hat{n}_3 \tag{4.22}$$

$$P_3 = C_i + \hat{n}_1 \left(\sqrt{f^2 (\hat{n}_1 \cdot \hat{n}_2)^2 - (f^2 - r_\varepsilon^{\ 2})} - f\hat{n}_1 \hat{n}_2 \right) \tag{4.23}$$

$$Q_i = C_i + r_\varepsilon \cdot \hat{n}_1 + |(Q_i - C_i) \times \hat{n}_1| \cdot \hat{n}_2 \tag{4.24}$$

$$d = |(Q_i - C_i) \times \hat{n}_3| + r_\varepsilon \tag{4.25}$$

After these boundary positions are expressed as a function of tool position C_i, feedrate, and tool geometries, they can be calculated easily and fast for any position of the tool on the toolpath. Based on this, the Green's theorem based analytical equations can be applied to extract the parameters of each *teF*.

4.5.4 Form Invariant Machining Feature (*fiF*)

A *fiF* is defined when the geometry of the *teF* changes in a predictable manner. There are four cases obtained from different combinations of line/arc workpiece boundary edges and line/arc toolpaths. The workpiece boundary Q_eQ_s and toolpath T_eT_s can be of types line-line (not parallel), line-arc, arc-line and arc-arc which are shown in Figure 4.15. Each case has two sub-cases where (a) illustrates Q_eQ_s intersecting the leading cutting edge of the tool, and (b) illustrates Q_eQ_s intersecting the tool's circular nose. In the case of one workpiece boundary edge intersecting both the straight tool edge and the circular tool edge along a toolpath, it can be separated into two single cases (a) and (b), and then dealt with separately.

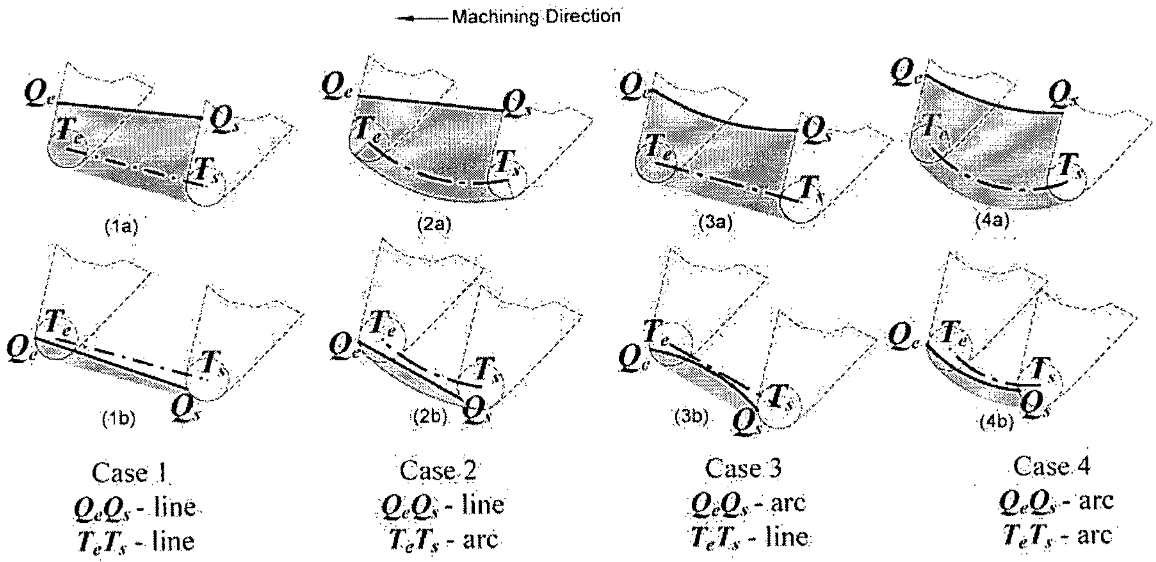


Figure 4.15: Four Types of Form Invariant Feature *fiF*

Similar analytical formulae have been developed that capture the intersection boundary points of the cutter, which are used to define the boundary of the *teF* at that location along the toolpath for a *fiF*. When the toolpath is a line segment, these formulae are calculated for each of the cases in the same way as identified above for the *giF*, i.e., the equations are the same as (4.17 ~ 4.25), except Q_i is different in equation (4.24) if the workpiece boundary is a circular edge. When the toolpath is an arc segment, since the feed direction varies at successive feed steps, these analytical expressions must be applied at each location along the tool path for the *teF* to be extracted. The formulae for arc toolpath as shown in Figure 4.16 are as follows.

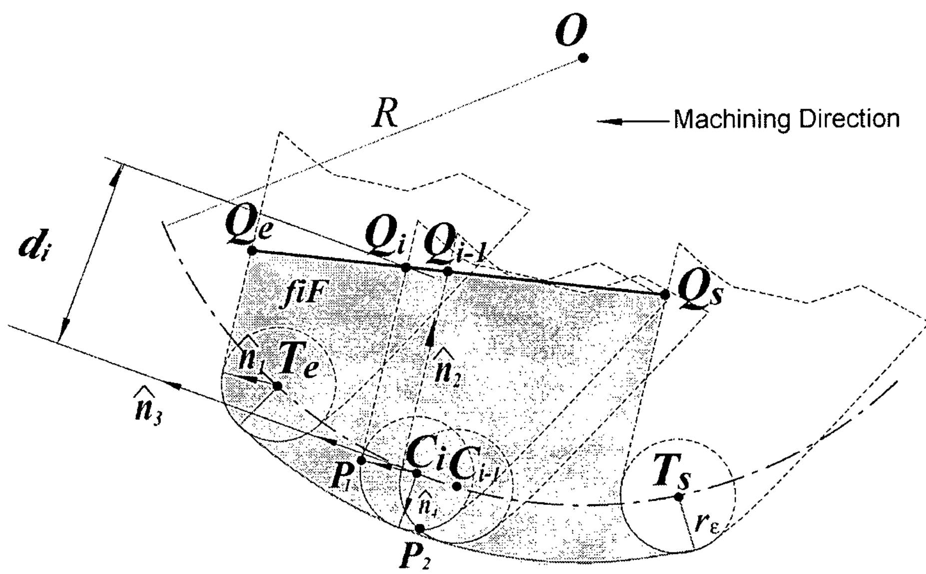


Figure 4.16: *teF* Extraction from *fiF*

$$P_1 = C_i + r_\varepsilon \cdot \hat{n}_1 \tag{4.26}$$

$$P_2 = C_i - R\sin(\frac{f}{2R}) \cdot \hat{n}_5 + \sqrt{r_\varepsilon^{\ 2} - R^2 \sin^2(\frac{f}{2R})} \cdot \hat{n}_6 \tag{4.27}$$

$$\text{where } \hat{n}_5 = \frac{C_i - C_{i-1}}{|C_i C_{i-1}|} = [-\sin(t - \frac{f}{2R}), \cos(t - \frac{f}{2R})]$$

$$\hat{n}_6 = [-\cos(t - \frac{f}{2R}), -\sin(t - \frac{f}{2R})]$$

$$P_3 = C_i + v \cdot \hat{n}_1$$

$$v = \sqrt{(l \cdot \hat{n}_1 \cdot \hat{n}_5)^2 - l^2 + r_\varepsilon^{\ 2}} - (l \cdot \hat{n}_1 \cdot \hat{n}_5) \tag{4.28}$$

$$Q_i = C_i + r_\varepsilon \cdot \hat{n}_1 + |(Q_i - C_i) \times \hat{n}_1| \cdot \hat{n}_2 \tag{4.29}$$

$$d = |(Q_i - C_i) \times \hat{n}_3| + r_\varepsilon \tag{4.30}$$

In these formulations, O is the circular toolpath center, and R is the radius of the toolpath.

Toolpath can be a line segment or an arc, for any position C_i of a tool on a toolpath it can be expressed as follows:

$$C_i = \begin{cases} (1-t)T_s + tT_e & 0 \le t \le 1 \\ O + [R\cos(t) \quad R\sin(t)] & \alpha_1 \le t \le \alpha_2 \end{cases} \tag{4.31}$$

where T_s, T_e are the known start and end positions of the tool path, and α_1, α_2 are the corresponding parametric angles of T_s and T_e. Successive expressions for C_i using C_{i-1} can be obtained from the above equation,

$$C_i = \begin{cases} C_{i-1} + \Delta t(T_e - T_s), \textit{ line toolpath} \\ O + [V_x \cos(\Delta t) - V_y \sin(\Delta t) \quad V_y \cos(\Delta t) - V_x \sin(\Delta t)] \quad \textit{arc toolpath} \end{cases} \tag{4.32}$$

where $V = [V_x \; V_y] = C_{i-1} - O$, O is the circular tool path center.

The tool increment is defined by $\Delta t = \begin{cases} f/L, & line \\ f/R, & arc \end{cases}$, where $L = |T_e T_s|$ is the line tool path length, and R is the radius of tool path.

Using the previous expression, the tool position C_i is easy to calculate from the previous position C_{i-1}. From this, boundary positions P_i can also be expressed using P_{i-1}. It can be seen that these positions of a *teF* do not need to be calculated from beginning at each feed step, further speeding up computation.

In conclusion, from the depth of cut and feedrate, the *teF* type is identified from the classification as shown in Figure 4.7. After all the boundaries are obtained as described above, the areas and centroids can be calculated using a Green's Theorem-based analytical formulation.

Furthermore, by using the boundary expressions in terms of C_i and by rearranging the Green's Theorem-based area equations, the areas can be expressed as a function of the toolpath parameter t (see equation (4.31)), $A = F(t)$, where $t = i \cdot \Delta t$, i is the feed step number and Δt is a tool position increment. With this expression, the area can be directly predicted with respect to the feed step along the toolpath. The details for deriving the boundary positions are shown in Appendix C.

4.6 Hybrid Analytical, Solid Modeler and Feature-Based Methodology

The hybrid methodology developed in this research merges the Green's Theorem-based analytical formulation, the Feature-based boundary identification methodology, and the Solid Modeler solutions together, to provide an efficient and complete mechanism for extracting *teF* parameters during the machining of each tool pass. A flowchart of this hybrid methodology is presented in Figure 4.17. Details of each step have already been given in the above sections. The steps of this Hybrid methodology are briefly described here. Results from the Full Solid Modeler Methodology and this Hybrid approach will be generated and compared using an industrial example in the next section.

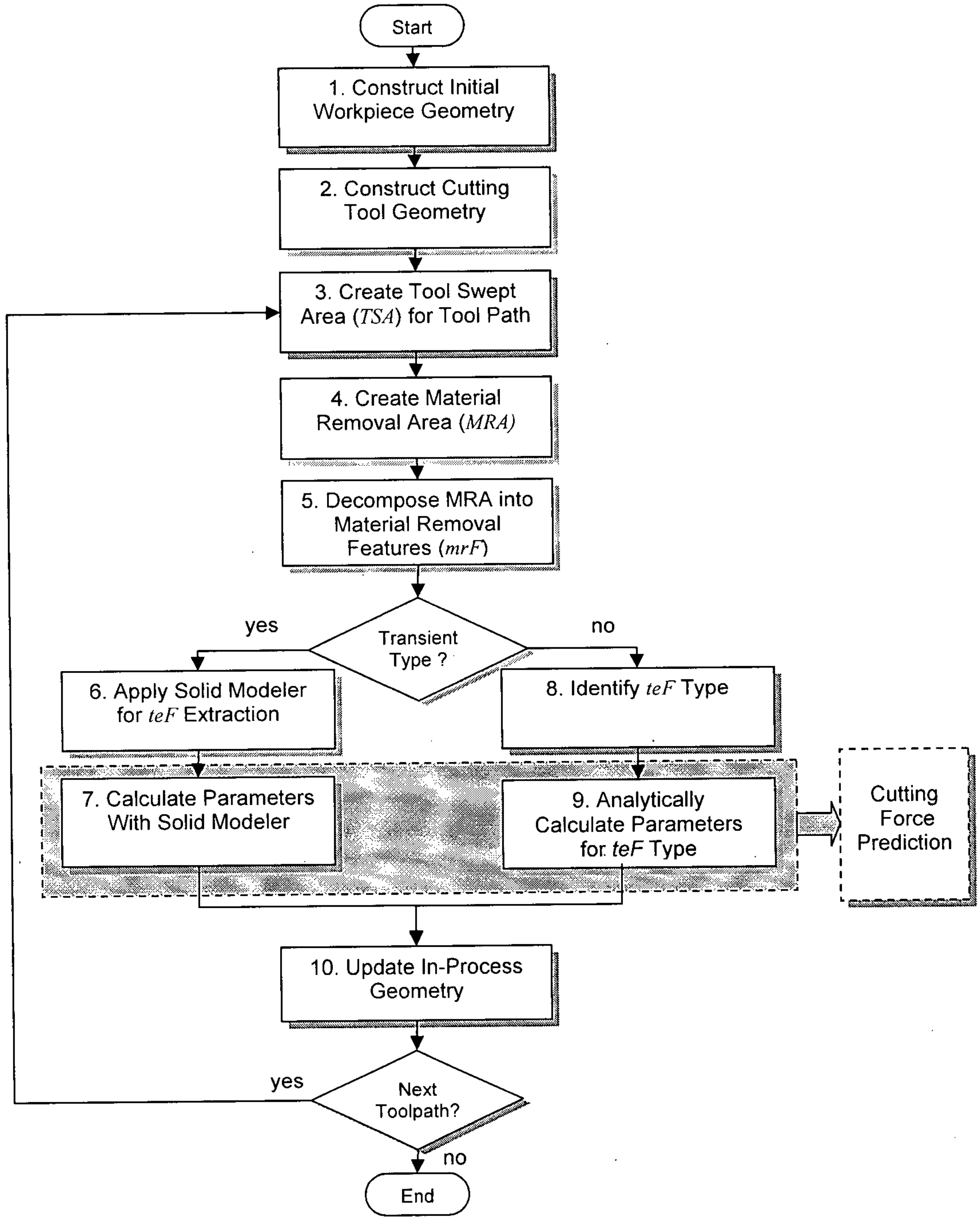


Figure 4.17: Hybrid *TWE* Extraction Methodology

As with the full Solid Modeler Solution, after the initial two steps where 2D models of the workpiece and the cutting tool geometry are created in the modeling environment[2], the *TSA* for the first tool path is generated. This is used to create a material removal area (*MRA*) for the tool path by performing a regularized Boolean intersection between the *TSA* and the workpiece (Step 4). In Step 5, this *MRA* is decomposed into removal features each belonging to one of the three types described in the previous section. Transient material removal features are processed to extract their constituent *teFs* using generic functions in the solid modeler (Step 6). *teF* parameters (areas and centroids) to be used in cutting force prediction are also extracted using the generic property evaluation functions of the modeler (Step 7). Non transient features are differentiated into *giF* and *fiF* types (Step 8) and analytical techniques based on the *teF* classification in Figure 4.7 are applied to calculate the area and centroidal parameters (Step 9). At this point a single toolpath has been processed for its engagement geometry. Before proceeding to the next toolpath the in-process workpiece must first be updated. This is accomplished in Step 10 by performing a Boolean subtraction between the current in-process workpiece and the *TSA* for the just completed tool path. These steps are repeated until all tool paths have been processed.

This hybrid methodology combines the generality of solid modeler-based functionality for handling transient engagement conditions with analytical solutions that enhance efficiency for regions where the engagement changes in a predictable and continuous way. Features are used to help in the formalization of the methodology.

---

[2] In this research the ACIS solid modeler a Dessault Systemes product is used to model and manipulate geometry

4.7 Implementation and Validation

The implementation and validation of the hybrid methodology is described in this section. An aerospace turned component shown in Figure 4.18 is used for this purpose. Two solutions (full solid modeler and hybrid) for extracting the *teFs* and their parameters are implemented separately in Visual C++ using the ACIS 3D modeling kernel and toolkit on a Windows Pentium4, 2.6GHz/512Mb, XP Workstation. Display and interaction with the in-process model utilizes the HOOPS 3dGS computer graphics database. Parametrized *teFs* are extracted at each feed step along a tool path. In addition, it is necessary to generate the tool swept area for each tool path and to subtract this from the in-process workpiece for toolpath *i* to prepare the workpiece for toolpath *i+1*. These areas are accordingly subtracted using Boolean function calls. Simulations of the machining for various tools and tool paths on the aerospace component are shown in Figure 4.19. Figure 4.20 illustrates examples of *giFs* , *fiF*, and *trFs* generated for one tool pass as part of the solution.

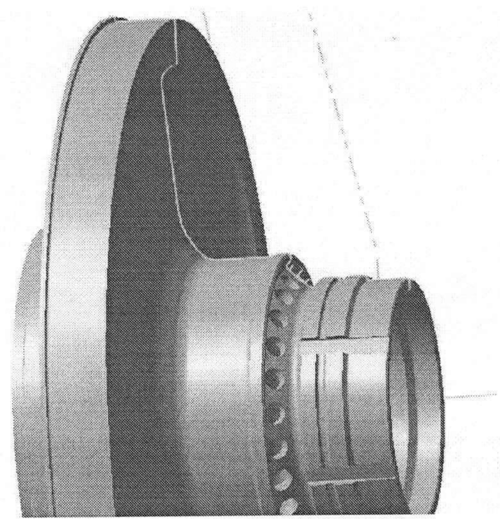


Figure 4.18: An Aerospace Turned Component Model

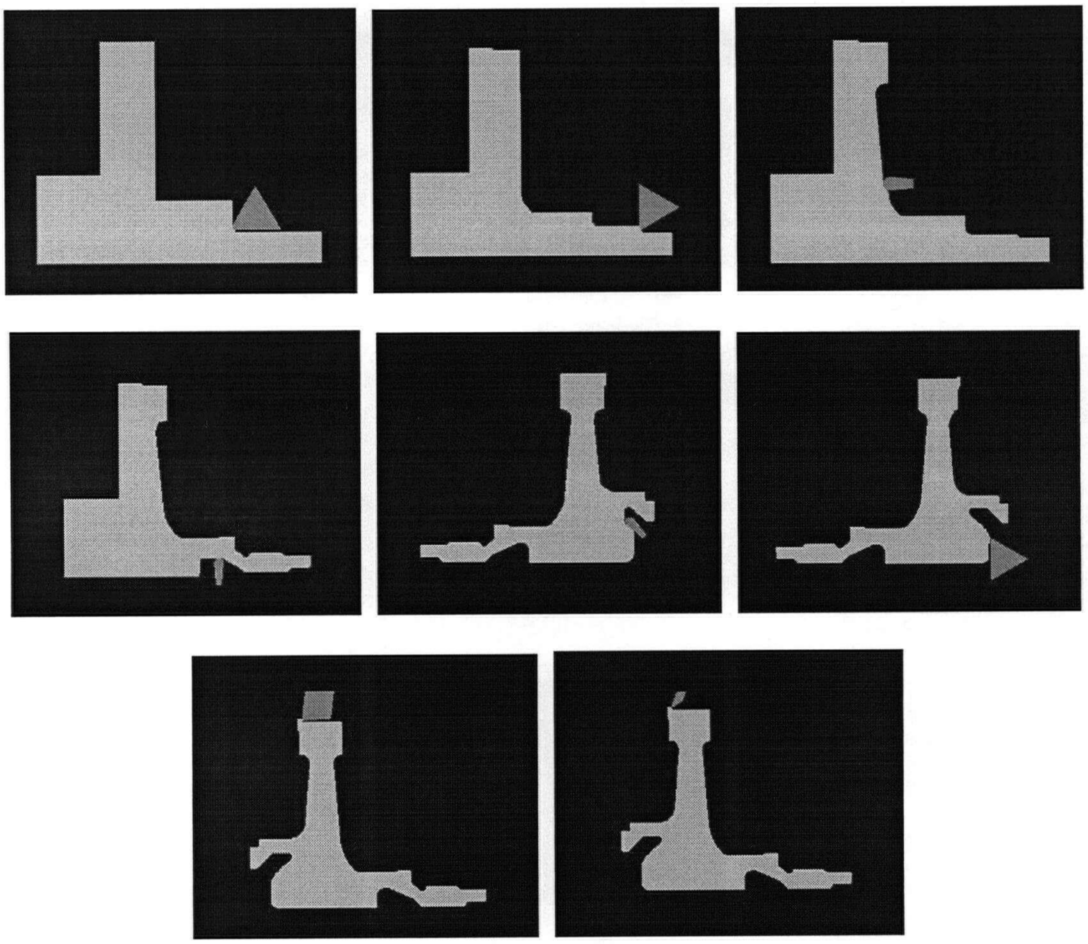


Figure 4.19: Simulation of the Machining for Various Tool Paths on Turning Part

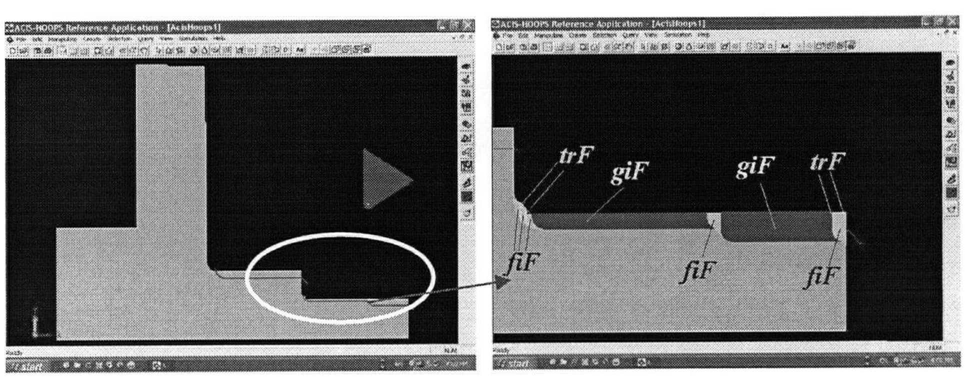


Figure 4.20: Extracted Material Removal Features for the Turned Part

		Full Solid Modeler Solution	Hybrid, Feature Based, Analytical Solution
Total Simulation Times (secs.)		172.094	14.984
Total Number of Intersection Area Calculation		4613	1503
Number of Intersection Area Calculation in *giFs*		3133	23
Times (secs.)	One Example Toolpath	5.360	0.058
	Single Intersection	0.020	0.00017
Accuracy	One Example of A (mm^2)	0.064444	0.06444
	L (mm)	1.208196	1.208196
	Effective Angle (red)	0.777640	0.777640

Table 4.2: Simulation Times and accuracy for Two Solutions

The computation times and accuracy for both solutions are listed in the above table. The feedrate used in this simulation is 0.126 mm/rev. From the comparison of total simulation times it can be seen that the pure solid modeler solution is about an order of magnitude slower than the hybrid solution. This is because Boolean operations need to be performed at each feed step. This can also be seen from the total number of intersection area calculations (4613) that must be performed. Moreover, the generic nature of area calculations performed by a solid modeler requires the use of a general purpose numerical solver. It is to be expected that the calculation time for this would be longer than an equivalent analytical solution, which involves exact integration over a small number of edges. Feature identification further speed up the calculations, especially for the geometry invariant features where only one intersection calculation is needed. Only 23 area calculations are needed for all the *giFs* identified in the test part. On the other hand, 3133 area calculations are required with the pure solid modeler solution. Furthermore, from table 2 it can be seen that the chip area, length and effective lead angle of both solutions are calculated as the same values. This verified the correctness of Green's Theorem-based analytical equations. And in conclusion, the both solutions can achieve the same the accuracy. It should be noted that the total simulation times indicated in Table 2 are actually greater than the sum of the intersection times or the sum of single toolpath times in the case of the hybrid solution. The reason for this is that the computation times for tool swept area calculations, material removal area subtraction, in-process workpiece updates, and other implementations for visualization are all included as the same as the full Solid Modeler solution. However, the time for solid modeling of both solutions is small compared to the total intersection and area calculation times, this makes the hybrid solution is still an order of magnitude faster than the full solid modeler solution.

The problem of robustness must be considered in full solid modeler solution, when applying large numbers of Boolean operations during simulation. In particular, Boolean operation errors can show up when boundary entities on two faces undergoing a Boolean operation overlap in a marginal way. Unlike design applications which are user interactive, these errors when they occur must be handled automatically. One approach to circumventing this problem is to represent the 2D workpiece and tool representation as thin 3D extrusions of slightly different thicknesses. Other strategies are incorporated as described in Yip-Hoi [1]. One of these is to perturb the position of the tool along the toolpath when an intersection operation fails. Contrarily, the hybrid approach is better in robustness since fewer intersections as indicated in the table translate into less opportunity for Boolean problems. From robustness point of view, the hybrid approach also made a big improvement compared to the full solid modeler solution.

4.8 Summary

A pure solid modeler solution and a hybrid analytical, solid modeler and feature based methodology for tool/workpiece engagement calculations in general turning processes are described in this research. The accuracy and computational efficiency are compared as shown in Table 2. It can be seen that the hybrid solution has significantly better computational efficiency than the pure solid modeler solution, while achieving the same accuracy. This is because Boolean operations which are applied at each feed step consume significant computational time. Whereas the latter hybrid solution which employees algebraic calculations has a significantly smaller processing time. Moreover, the identification and parametrization of geometric invariant features and form invariant features, eliminates large amounts of repetitive calculations, therefore leading to further improvements in the computation time.

As discussed in the assumptions section, one area for future work will consider deflection and dynamics which result in process induced variations in the geometry of the intersection area. The 2D model will need to be extended to a 3D model for capturing this effect. A 3D methodology is also necessary for capturing engagement conditions for non-symmetric parts due to the initial workpiece having previously machined non-turned features such as holes and slots. The feature identification needs to be extended to 3D volume features.

Chapter 5

Instantaneous Force Prediction for Contour Turning

5.1 Introduction

A substantial amount of research has been done to predict static turning forces for given depth of cut, feedrate, cutting speed, and tool geometry. A mechanistic model has emerged as a successful approach for cutting force prediction. The mechanistic model proposed by Atabey [1,2] presents the cutting forces in tangential and friction directions. Friction force is perpendicular to the cutting edge and passes through the gravity center of the uncut chip area. This model gives good prediction of force direction to simplify the force calculation. Particularly for the force prediction in dynamics of turning when the uncut chip area becomes irregular, the force model remains the same.

However, the aforementioned model is capable of computing mechanics parameters for simple workpiece geometry at one feed step, where feed is in the direction of the spindle axis and the depth of cut is constant. In contour turning, the feed direction varies along the toolpath. Therefore, the forces are predicted in the local coordinates with respect to feed direction. Local tangential force (F_t) is in the same direction with global tangential force ($\mathbf{F_t}$); local feed Force (F_f) and local radial force (F_r) need to be resolved and summed in the global Feed $\mathbf{F_f}$ (spindle axis, called Z) and global Radial $\mathbf{F_r}$ (X axis) directions, which will be discussed in Section 5.3, for power, torque calculation, and machine constrain-based optimization of the turning process. Deflection is predicted based on local feed and radial forces in local coordinates.

In addition to feed direction, instantaneous depth of cut, uncut chip area, chip-cutting edge contact length and effective lead angle change along the contour path or when the workpiece geometry varies. The cutting force prediction requires dynamic identification of removed chip shape at each feed increment to simulate part turning. The tool-workpiece intersection is identified through a geometric and solid modeling system, which was presented in Chapter 4.

The rest of this chapter is organized as follows. Section 5.2 presents details of the adopted mechanistic force model, which is slightly modified to improve force prediction efficiency for virtual turning. The mechanistic model of contour turning is presented in section 5.3. Section 5.4 introduces a new method to identify the mechanistic cutting coefficients from the orthogonal cutting database. This is followed by experimental validation on an Aluminum test part in Section 5.5. The chapter ends with conclusions and recommendations for future research.

5.2 Mechanistic Model in Simple Turning

In the mechanistic model proposed by Atabey [1,2], the turning tool has a nose radius, and the cutting forces are represented by the tangential force (F_t) and friction force (F_{fr}), shown in Figure 5.1.

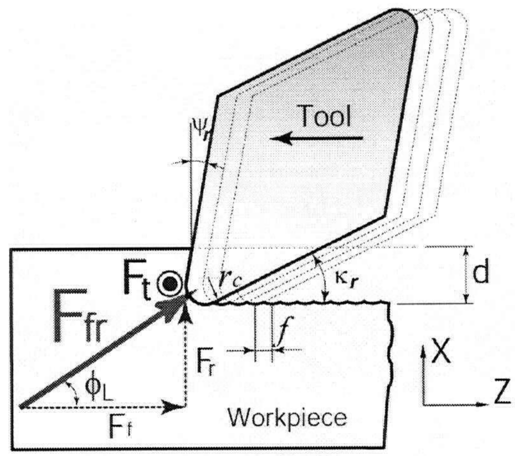


Figure 5.1: Mechanistic Force Model

Since the chip thickness distribution at each point along the cutting edge contact point is different and dependent on the tool nose radius (r_ε), side and end cutting edge angle, feedrate *(f)* and radial depth of cut (*d*), the distribution of the force along the cutting edge-chip contact zone also varies. At any contact point, the differential cutting forces are modeled as a function of local chip load (*dA*) and chip-cutting edge contact length *(dL*c), See Figure 5.2.

$$\begin{aligned} dF_t &= dF_{tc} + dF_{te} = K_{tc} \cdot dA + K_{te} \cdot dL_c \\ dF_{fr} &= dF_{frc} + dF_{fre} = K_{frc} \cdot dA + K_{fre} \cdot dL_c \end{aligned} \tag{5.1}$$

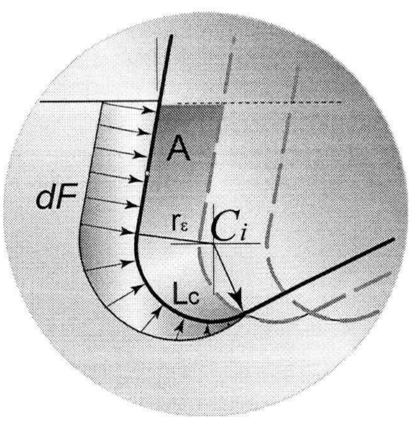


Figure 5.2: Distribution of Friction Force along Cutting Edge

The direction of each differential tangential force is perpendicular to the 2D cross-section of the workpiece and the tool, as shown in Figure 5.1. However, the direction of differential friction force varies in different regions of the uncut chip. In the tool straight cutting edge region (Region 2 shown in Figure 5.3), the chip thickness does not change and the effective lead angle is the same as the side cutting edge angle. The direction of each differential force remains the same, i.e., perpendicular to the same straight cutting edge, as well as the magnitude. While in the tool nose curve region (Region 1 shown in Figure 5.3), the differential chip area changes continuously, and the friction force acts perpendicular to the cutting edge segment for each differential element, it can be predicted by assuming that each component of the friction force passes through the gravity center of each related region (Figure 5.3). The friction force component of each region is added up vectorially to find the total friction force (F_{fr}).

Due to the different mechanics at the tool nose curve and straight cutting edge regions, the total tangential force (F_t) and friction force (F_{fr}) are modeled as follows:

$$\begin{aligned} F_t &= F_{tc} + F_{te} = K_{tc} \cdot A + K_{te} \cdot L_c \\ F_{fr} &= F_{frc1} + F_{frc2} + F_{fre} = K_{frc1} \cdot A_1 + K_{frc2} \cdot A_2 + K_{fre} \cdot L_c \end{aligned} \tag{5.2}$$

The cutting coefficients $K_{tc}, K_{frc1}, K_{frc2}, K_{te}, K_{fre}$ are obtained from cutting tests and curve-fitting technics. They are non-linear functions of chip load, chip-cutting edge contact length and cutting speed.

Friction force F_{fr} is considered to consist of two cutting force components F_{frc1} and F_{frc2} corresponding to the cutting forces in region 1 and region 2, which are associated with uncut chip areas A_1 and A_2. Later, F_{fr} is resolved into the feed (F_f) and radial directions (F_r) with respect to the resultant effective lead angle ϕ_L. See Figure 5.3.

$$F_r = F_{fr} \cdot \sin(\phi_L)$$
$$F_f = F_{fr} \cdot \cos(\phi_L) \tag{5.3}$$

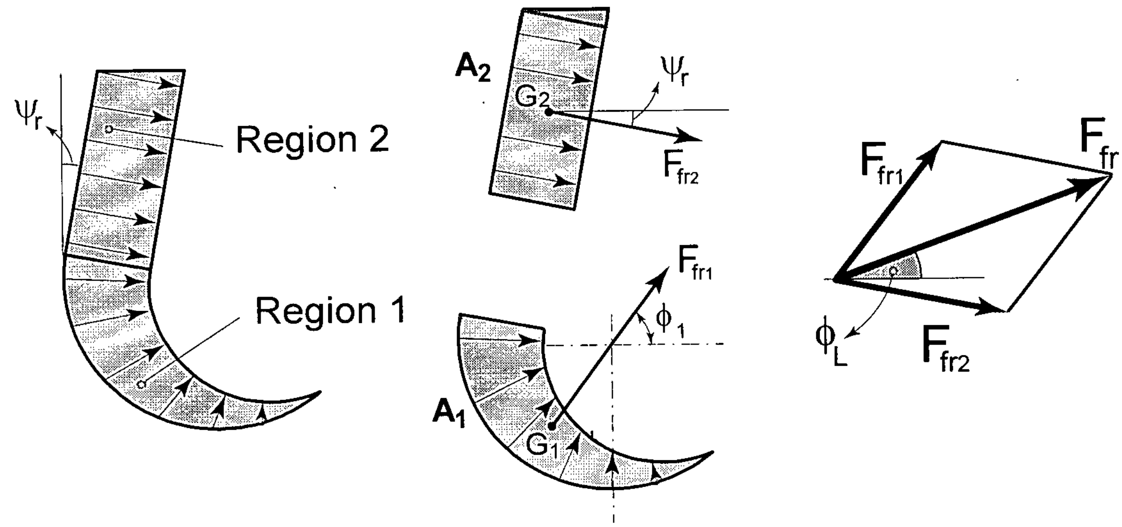


Figure 5.3: Friction Forces and Effective Lead Angle (ϕ_L)

In the original mechanistic force model, the calculation of effective lead angle (ϕ_L) is $\phi_L = \dfrac{\phi_1 A_1 + \psi_r A_2}{A_1 + A_2}$. For relatively large radial depth of cut, the effective lead angle tends to approach the side cutting edge angle (ψ_r). However, since the effective lead angle defines the direction of total friction force (F_{fr}), which is the resultant force evaluated from the two regions (F_{fr1}, F_{fr2}), F_{fr1} and F_{fr2} in the tool nose region and straight cutting edge region are assumed to contribute to the direction of F_{fr}. Therefore, the ϕ_L, which was considered to be dependent only on the geometric information in the original model, showed some discrepancies between measured and predicted effective lead angles. Atabey introduced modification factor K_m, which is a linear function of chip length (L_c) and cutting speed (V), to correct the effective lead angle calculation. The modified effective lead angle is $\phi = K_m(L_c, V) \cdot \phi_L$. Later, the radial force and the feed force are calculated using this lead angle.

$$\begin{aligned} F_r &= F_{fr} \cdot \sin(\phi) \\ F_f &= F_{fr} \cdot \cos(\phi) \end{aligned} \tag{5.4}$$

To eliminate using the modification factor and to minimize the discrepancy caused by ϕ_L, a slight modification to the original model is presented as follows. F_{fr1} and F_{fr2} are calculated in each region separately instead of calculating the resultant friction force F_{fr}. The forces are resolved by using the effective lead angle of each region and summed up to form resultant F_r, F_f as shown in Figure 5.4. From equation (5.2), F_{fr1} and F_{fr2} are obtained as

$$\begin{aligned} F_{fr1} &= K_{frc1} \cdot A_1 + K_{fre} \cdot L_{c1} \\ F_{fr2} &= K_{frc2} \cdot A_2 + K_{fre} \cdot L_{c2} \end{aligned} \tag{5.5}$$

The edge coefficient K_{fre} is assumed to be constant in both regions, and the radial force and feed force in each region are calculated as follows.

$$\begin{aligned} F_{r1} &= F_{fr1} \cdot \sin(\phi_1) \\ F_{f1} &= F_{fr1} \cdot \cos(\phi_1) \end{aligned} \tag{5.6}$$

where ϕ_1 is the gravity vector angle shown in Figure 5.4

$$\begin{aligned} F_{r2} &= F_{fr2} \cdot \sin(\psi_r) \\ F_{f2} &= F_{fr2} \cdot \cos(\psi_r) \end{aligned} \tag{5.7}$$

where ψ_r is the side cutting edge angle. Final radial and feed force is:

$$\begin{aligned} F_r &= F_{r1} - F_{r2} \\ F_f &= F_{f1} + F_{f2} \end{aligned} \tag{5.8}$$

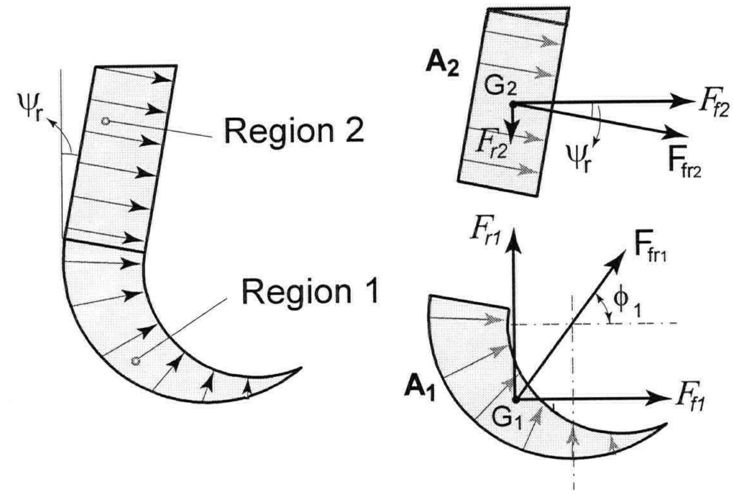


Figure 5.4: Feed, Radial Forces in Each Region

As a result, the modification factor is not used and the predicted radial and feed forces match well with the measured data as shown in Section 5.5.

5.3 Prediction of Cutting Forces in Contouring Turning

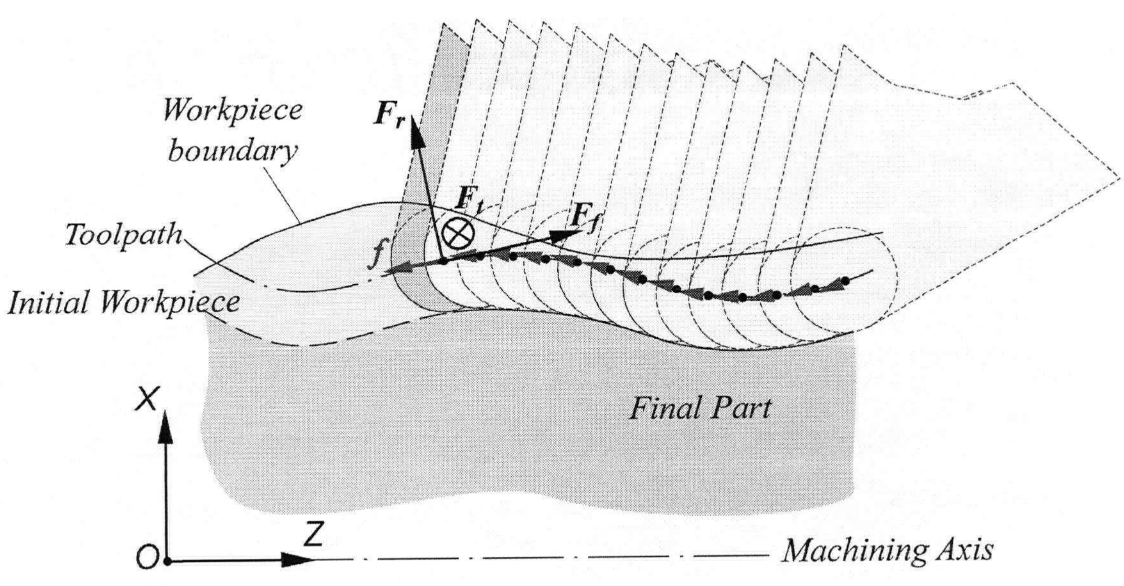


Figure 5.5: Contour Turning

As shown in Figure 5.5, the forces predicted using the mechanistic force model described in the previous section are the local forces with respect to instantaneous feed direction at each feed step. To apply the mechanistic model to contour turning, the local forces F_t, F_r, and F_f need to be projected to global XYZ directions. Tangential force F_t is in the same direction as the global tangential force $\mathbf{F_t}$. Radial and Friction forces F_r, F_f at each feed step need to be resolved in global machine axes (X, Z). The projected forces in global axes are called global radial ($\mathbf{F_r}$) and feed ($\mathbf{F_f}$) forces, respectively. The global forces $\mathbf{F_t}$, $\mathbf{F_r}$, and $\mathbf{F_f}$ at each machining step are shown in Figure 5.6.

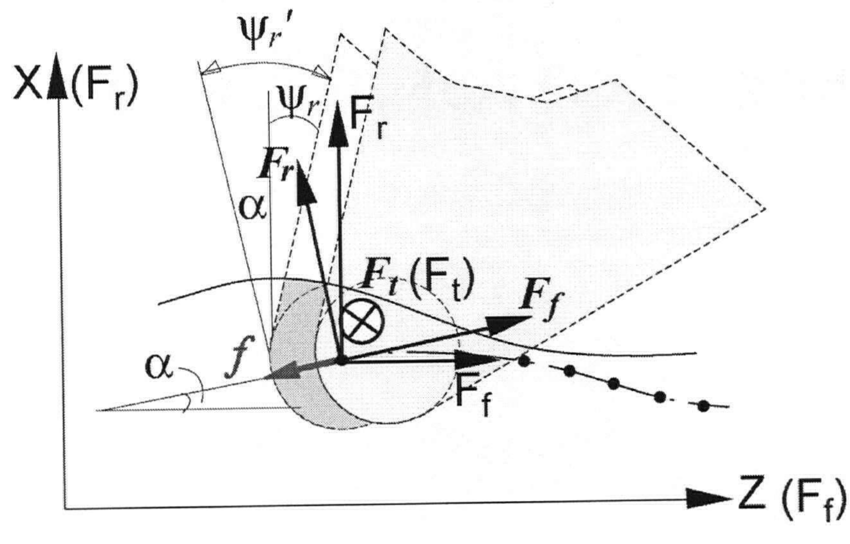


Figure 5.6: Force Prediction of Contouring Turning

The global cutting forces at any feedrate step are:

$$
\begin{aligned}
\mathbf{F_t} &= F_t \\
\mathbf{F_r} &= F_f \cdot sin(\alpha) + F_r \cdot cos(\alpha) \\
\mathbf{F_f} &= F_f \cdot cos(\alpha) + F_r \cdot sin(\alpha)
\end{aligned}
\tag{5.9}
$$

Where, α is the angle between the instantaneous feed direction and the machining axis shown in Figure 5.6. Using equation (5.9) the instantaneous global $\mathbf{F_t}$, $\mathbf{F_r}$, and $\mathbf{F_f}$ are predicted at each machining feed step. When all these three forces are calculated at all feed steps along the toolpath, the whole cutting process is simulated. Based on forces, powers, torques, and the deflections, the machining process can be optimized.

However, it can be seen that the global force calculation method of contour turning (equation (5.9)) is not efficient enough in terms of projection twice. In the local coordinates with respect to the instantaneous feed direction, the local side cutting edge angle (ψ_r') varies with α, i.e., $\psi_r' = \psi_r + \alpha$, where ψ_r is the fixed side cutting edge angle of the tool along machining. Local friction forces (F_{fr1}, F_{fr2}) need to be projected to the local radial (F_r) and feed (F_f) (equations (5.4~5.7)) and then be projected again to global radial ($\mathbf{F_r}$) and feed ($\mathbf{F_f}$) forces (equation (5.9)).

To increase computational efficiency, mathematical simplification is achieved by manipulating the above formulations. From equations (5.5~5.7), the local F_r and F_f are expressed as the combination of the friction forces in two regions,

$$\begin{aligned} F_r &= F_{r1} - F_{r2} = F_{fr1} \cdot \sin(\phi_l') - F_{fr2} \cdot \sin(\psi_r') \\ F_f &= F_{f1} + F_{f2} = F_{fr1} \cdot \cos(\phi_l') + F_{fr2} \cdot \cos(\psi_r') \end{aligned} \tag{5.10}$$

where ϕ_l' and ψ_r' are the local gravity angle and local side cutting edge angle at any feed step. Similar to $\psi_r' = \psi_r + \alpha$, the global gravity angle is $\phi_l = \phi_l' + \alpha$.

Substituting F_r, F_f, ϕ_l' and ψ_r' in equation (5.10), the global forces in equation (5.9) are obtained as

$$\begin{aligned}\mathbf{F_t} &= F_t \\ \mathbf{F_r} &= F_{fr1}\sin(\phi_l) - F_{fr2}\sin(\psi_r) \\ \mathbf{F_f} &= F_{fr1}\cos(\phi_l) + F_{fr2}\cos(\psi_r)\end{aligned} \tag{5.11}$$

Where ϕ_l is the angle of a vector that goes through the gravity center of tool nose region and points to the tool nose center in global coordinates. The global force formulation shown in equation (5.11) can significantly simplify the force calculation during arbitrary contour turning, because the side cutting edge angle (ψ_r) remains unchanged, and only the global gravity center angle (ϕ_l) is calculated directly in global coordinates without projection twice at each feed step.

However, it must be noted that the force expressions for contour turning are based on the assumption that the changes in side cutting edge angle along the contour toolpath do not affect cutting coefficients significantly. In this research this assumption is used to simplify the force calculation based on the experimental results shown in Section 5.5. More cutting tests are needed at different contour conditions to generalize the comment. Moreover, the force expression (equation (5.11)) is merely the mathematical equations without physical meaning.

As presented in the previous section, the cutting force coefficients used in the equations (5.2~5.4) are identified from the cutting tests or from the orthogonal database (which will be discussed in the next section). Instantaneous tool–workpiece intersection information (uncut chip area (A), contact chip length (L_c), global gravity center of the tool nose region (ϕ_l), and tool geometry information (ψ_r, κ_r, r_ε) are generated from the tool-workpiece engagement model described in Chapter 4. The instantaneous global forces $\mathbf{F_t}$, $\mathbf{F_r}$, $\mathbf{F_f}$, power, and torque of all machining steps are predicted using the extended mechanistic force model. The comparison between the predicted forces and the measured forces of a test part will be presented in Section 5.5.

5.4 Mechanistic Cutting Coefficient Evaluated form the Orthogonal Cutting Database

The mechanistic cutting coefficients K_{tc}, K_{frc1}, K_{frc2}, K_{re}, and K_{fre} are identified from turning tests for each tool/workpiece combination, and they are modeled as non-linear functions of the A, L_{c1}, L_{c2} and V:

$$\begin{aligned} K_{tc} &= b_0 A^{b_1} V^{b_2} \\ K_{frc1} &= m_0 L_{c1}^{m_1} V^{m_2} \quad [\text{N/mm}^2] \\ K_{frc2} &= n_0 L_{c2}^{n_1} V^{n_2} \end{aligned} \tag{5.12}$$

where b_0, b_1, b_2, m_0, m_1, m_2, n_0, n_1, and n_2 are empirical constants that are evaluated from the experimentally measured force data using the least-square method.

There are three advantages to the mechanistic cutting coefficient identification method. First, they are expressed as a function of a few geometric variables with simple expressions. Second, the empirical constants of mechanistic coefficients are valid for all cutting conditions. Third, K_{tc} expresses the tangential coefficient of the whole uncut chip area, and K_{frc1} and K_{frc2} express the friction coefficients in two regions of the uncut chip area. These expressions make the force prediction simple because the uncut chip area is only separated into two regions at each feed step. The simplicity and coverage of multiple cutting conditions lead to efficient cutting force computation in contour toolpaths where the chip geometry changes continuously. Contrarily, the orthogonal to oblique transformation force prediction method digitizes the cutting edge into small micro-elements, which requires at least an order of magnitude more computation at each feed step.

On the other hand, the disadvantage of the mechanistic method is that it is only valid for one tool geometry and workpiece combination. Hence, each tool geometry must be calibrated through cutting tests.

The orthogonal to oblique transformation force prediction method [7] uses fundamental material properties, such as shear angle, shear stress, and friction angle to determine the oblique cutting coefficients. The advantage of this method is that it is valid for a range of cutting tool geometries. Therefore, once an orthogonal cutting database is established, the oblique cutting coefficients of any tool geometry of a material can be identified. However, when a tool has a nose radius, the cutting edge must be considered as an assembly of the straight oblique cutting edge elements. This force method is less efficient in contour turning, because there are a large number of different tool-workpiece intersections along the toolpaths and in each instantaneous uncut chip area, the cutting edge has to be discretized to small segments, and all the differential cutting coefficients need to be calculated through orthogonal to oblique transformation [7]. Moreover, the method requires that the rake face of the cutting tool is uniformly flat (i.e., without any chip breaking or contact reduction grooves).

As an alternative to the pure mechanistic or the pure orthogonal to oblique transformation at each tool-workpiece engagement, the mechanistic cutting coefficients are directly evaluated from the orthogonal cutting database.

An oblique tool with curved nose and smooth rake face are considered, and the cutting forces are predicted in sets of given cutting conditions using orthogonal to oblique transformation by digitizing the cutting edge. The predicted forces are used like the measured forces in fitting the mechanistic cutting force coefficients.

The following example is given to illustrate the above identification method. The existing orthogonal cutting database of Aluminum 6061-T6 and the tool geometry used are shown as follows.

- **Existing Orthogonal Cutting Database of Aluminum 6061-T6**

Edge cutting coefficients: (N/mm)	$K_{te} = 24.416 - 1.9907e^{-5} \cdot V^2 + 0.045502 \cdot V - 0.16499 \cdot RN$ $K_{fe} = 9.8695 - 6.0897e^{-5} \cdot V^2 + 0.13283 \cdot V - 0.00162 \cdot RN$ $K_{re} = 0$
Shear stress: (N/mm^2)	$\tau_s = 244.49 + 0.336 \cdot RN$
Shear angle: (degree)	$\phi_c = 12.758 + 53.650 f + 0.0073698 \cdot V + 0.2972 \cdot RN$
Friction angle: (degree)	$\beta_a = 53.473 - 4.2403 f + 2.5759e^{-5} \cdot V^2 - 0.069330 \cdot V + 0.297 \cdot RN$

f – feedrate (mm/rev), RN - Side rake angle (deg), V – Cutting speed (m/min)

- **Tool Geometry**

Tool nose radius: (mm)	$r_\varepsilon = 0.7874$
Side cutting edge angle: (degree)	$\psi_r = 22.5$
End cutting edge angle: (degree)	$\kappa_r = 32.5$
Side and back rake angle (degree)	$\alpha_f = \alpha_p = 0$

- **Force Prediction Based on Orthogonal to Oblique Transformation**

As introduced in the literature review, three differential cutting force components can be expressed as follows:

$$
\begin{aligned}
dF_t &= dF_{tc} + dF_{te} = dK_{tc} \cdot b \cdot h + K_{te} \cdot b \\
dF_f &= dF_{fc} + dF_{fe} = dK_{fc} \cdot b \cdot h + K_{fe} \cdot b \\
dF_r &= dF_{rc} + dF_{re} = dK_{rc} \cdot b \cdot h + K_{re} \cdot b
\end{aligned} \tag{5.12}
$$

Where the oblique cutting coefficients are given in equation (5.13) refer to [4,7]. τ_s, β_n, and ϕ_n come from the orthogonal database, rake angle α_n, oblique angle i, and η come from the tool geometry.

$$
\begin{aligned}
dK_{tc} &= \frac{\tau_s}{\sin\phi_n} \cdot \frac{\cos(\beta_n - \alpha_n) + \tan i \tan\eta \sin\beta_n}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2\eta \sin^2\beta_n}} \\
dK_{fc} &= \frac{\tau_s}{\sin\phi_n \cos i} \cdot \frac{\sin(\beta_n - \alpha_n)}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2\eta \sin^2\beta_n}} \\
dK_{rc} &= \frac{\tau_s}{\sin\phi_n} \cdot \frac{\cos(\beta_n - \alpha_n)\tan i + \tan\eta \sin\beta_n}{\sqrt{\cos^2(\phi_n + \beta_n - \alpha_n) + \tan^2\eta \sin^2\beta_n}}
\end{aligned} \tag{5.13}
$$

The differential forces for the regions 1 of each uncut chip area in global X, Y and Z directions are identified in equation (5.14), region 2 and 3 are shown in equation (2.6-2.7), and the total integrated forces F_X, F_Y, F_Z are expressed in equation (5.15).

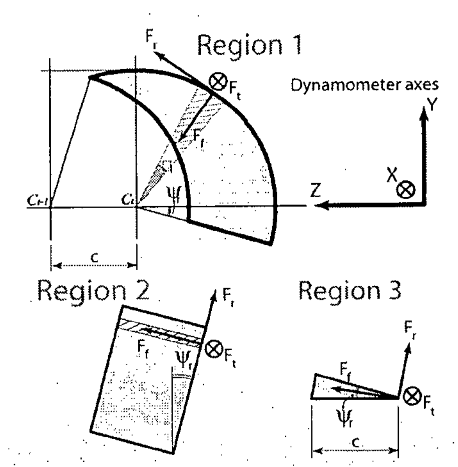


Figure 5.7: Orthogonal to Oblique Transformation

$$
\begin{aligned}
F_{x,1} &= \sum_{i=1}^{n} F_{t1,i} \\
F_{y,1} &= \sum_{i=1}^{n} (F_{f1,i} \sin\theta_i - F_{r1,i} \cos\theta_i) \\
F_{z,1} &= \sum_{i=1}^{n} (F_{f1,i} \cos\theta_i - F_{r1,i} \sin\theta_i)
\end{aligned}
\tag{5.13}
$$

$$
\begin{aligned}
F_x &= F_{x,1} + F_{x,2} + F_{x,3} \\
F_y &= F_{y,1} + F_{y,2} + F_{y,3} \\
F_z &= F_{z,1} + F_{y,2} + F_{z,3}
\end{aligned}
\tag{5.14}
$$

For given cutting speed (m/min) 250, 375, 500, feedrate (mm/rev) 0.05, 0.075, 0.1, 0.125, 0.15, 0.175, 0.2, 0.25, and the depth of cut (mm) 0.2, 0.4, 0.7, 1.0, 1.5, 2.0, 2.5, 3.0, the forces in total 216 cutting conditions are predicted. The predicted cutting forces (F_x, F_y, F_z) are regarded as the measured forces in the 216 different sets of cutting conditions. After Least-square curve fitting [1,2], the final mechanistic cutting coefficients are shown as follows.

- **Mechanistic Cutting Coefficients:**

K_{te}(N/mm)	K_{re}(N/mm)	K_{fe}(N/mm)	K_{fre}(N/mm)
48.873815	20.025611	43.391623	47.789728
$K_{tc} = 1863.82 \cdot A^{(0.0675)} \cdot V^{(-0.128997)}$ (N/mm^2)			
$K_{frc1} = 375.382 \cdot L_{c1}{}^{(-0.442)} \cdot V^{(-0.0511434)}$ (N/mm^2)			
$K_{frc2} = 241519.4 \cdot L_{c2}{}^{(0.397388)} \cdot V^{(-1.120065)}$ (N/mm^2)			

Where A (mm^2), L_{c1}, L_{c2} (mm), V (m/min)

From the previous cutting test validations, the pure mechanistic cutting coefficient identification method (least square techniques from the measured data) gives less than 10%

errors when enough measured forces are conducted, and the pure orthogonal to oblique transformation method also gives good prediction when discretized cutting edge elements are small enough. Therefore, the error of the predicted force, in which the mechanistic cutting coefficients are evaluated from the orthogonal database, remains the same order as the orthogonal to oblique transformation method. The comparison between the forces whose cutting coefficients are evaluated from orthogonal to oblique transformation and orthogonal database are shown as the following figures.

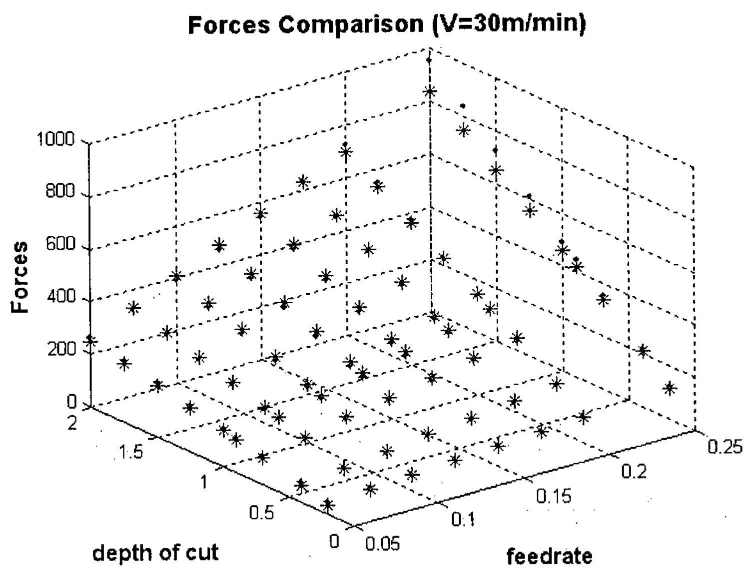


Figure 5.8: Tangential Force Predicted from Different Cutting Coefficient Identifications

Figure 5.8 shows forces predicted from different cutting coefficient identification methods. Dots represent the forces that are calculated using the orthogonal to oblique transformation method, and the stars represent the forces that are calculated using the mechanistic method, in which the mechanistic cutting coefficients are evaluated from the orthogonal cutting database.

The plot shows that the forces predicted from the two methods match well. The following figures also show the forces predicted using the two methods in different cutting conditions.

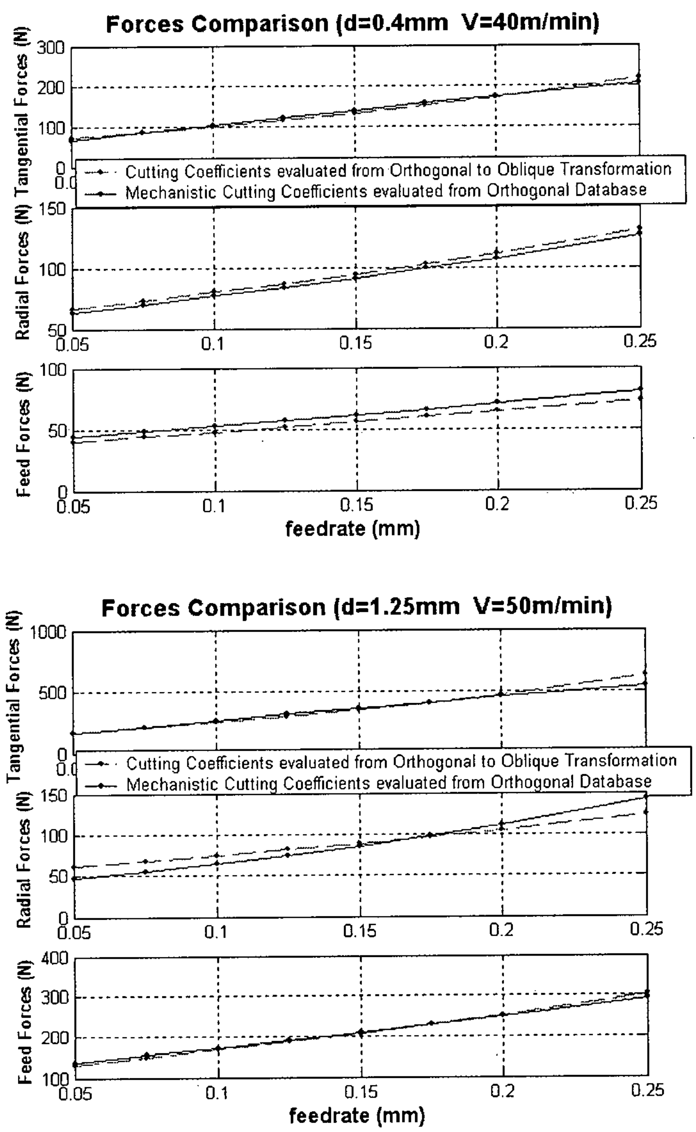


Figure 5.9: Forces Predicted from Different Cutting Coefficients

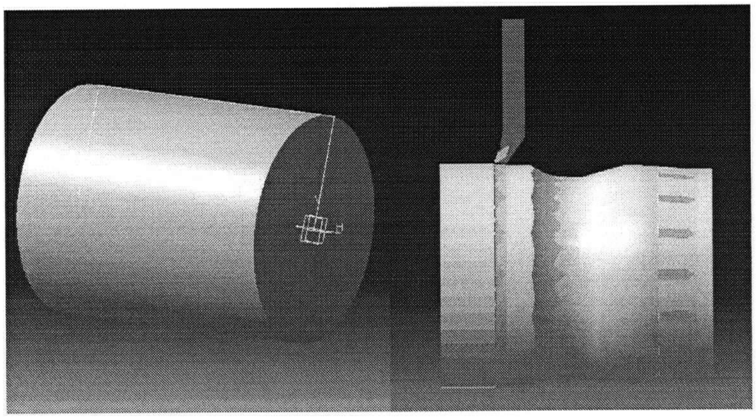
The above comparisons demonstrate that the mechanistic cutting coefficients evaluated from the orthogonal database provide good force prediction results. The forces predicted using these mechanistic coefficients are very close to the forces predicted from the orthogonal to oblique method. More cutting tests will be done to compare the forces that are calculated from different cutting coefficients. Depending on the accuracy of the well-known orthogonal to oblique approach and the mechanistic cutting coefficient identification method, it can be concluded that mechanistic cutting coefficients evaluated from the orthogonal database method is feasible and practical.

5.5 Experimental Validation for Contour Turning

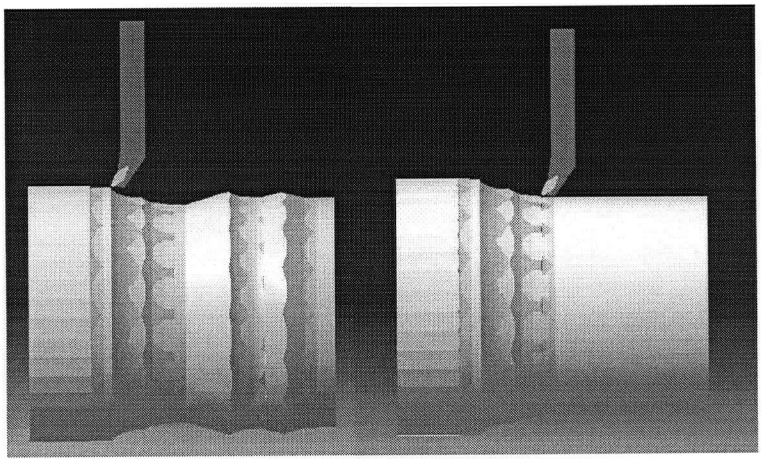
To test the tool-workpiece intersection and mechanistic force prediction model, an Aluminum 6061-T6 test part was machined on the Cincinnati Falcon 300 CNC turning center. A turning tool, P052.1 - Holder PT 135789, Insert PC 157838 with a nose radius of 0.7874mm, was used in the experiments. Kistler 9257B dynamometer and MalDaq 6.0 software were used to measure the cutting forces. The rake face of the tool was flat, hence it was possible to use the orthogonal to oblique transformation theory.

The predicted forces are based on the mechanistic force model of contour turning, in which the cutting force coefficients are predicted from the orthogonal database. Tool-workpiece intersection geometry was generated from the tool-workpiece engagement model. The cutting tests are designed as follows.

5.5.1 Cutting Test Design



Initial Workpiece First Cut



Second Cut Third Cut

Figure 5.10: Turning Process Plan of the Test Part

The turning operations to produce the final test part include three series of cuts. The first cut tests the correctness of the intersection geometries and the force results on the initial cylindrical workpiece with the contour toolpaths. The second cut tests those on the contour in-process workpiece, which is the resulting workpiece after the first cut, with the contour toolpaths. And the third cut tests the contour workpiece with the straight toolpaths. The comparisons between the measured and predicted cutting forces along the contour turning are presented in the following sections.

5.5.2 First Operation

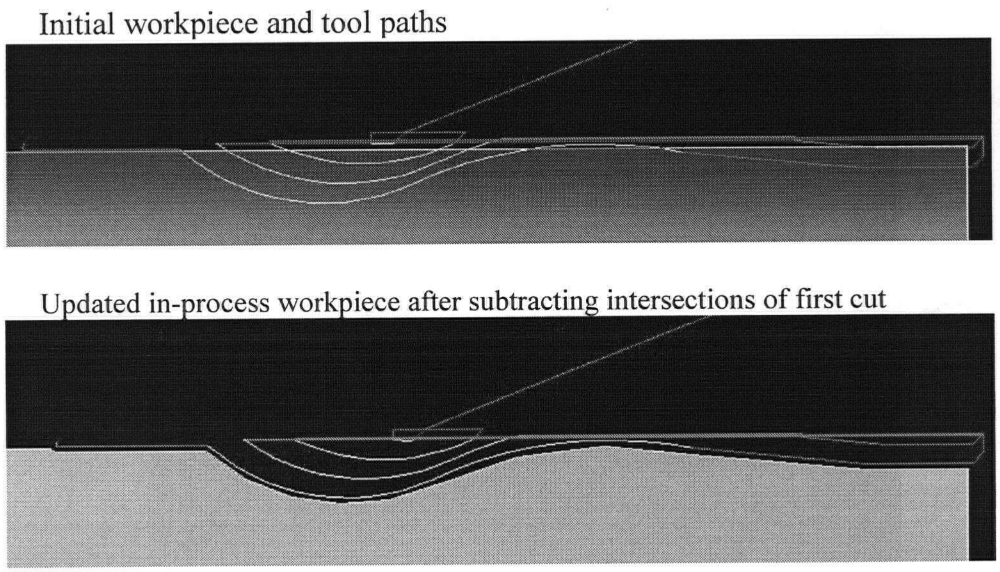


Figure 5.11: Tool Paths and Workpiece of First Cut

Figure 5.11 shows the designed initial workpiece and toolpaths in a CAD environment and the simulation result of the turning process in the developed tool workpiece intersection model of this research. The tool moves along the toolpaths, and tool-workpiece engagement at each machining step is captured and calculated. The material removal area (Boolean intersection between the tool swept area and the in-process Workpiece) is obtained and subtracted from the workpiece. Since the forces are proportional with the intersections as shown in equations (5.2~5.4), the material removal area of each toolpath and the corresponding predicted forces are shown in the following figures to verify the intersections and predicted forces.

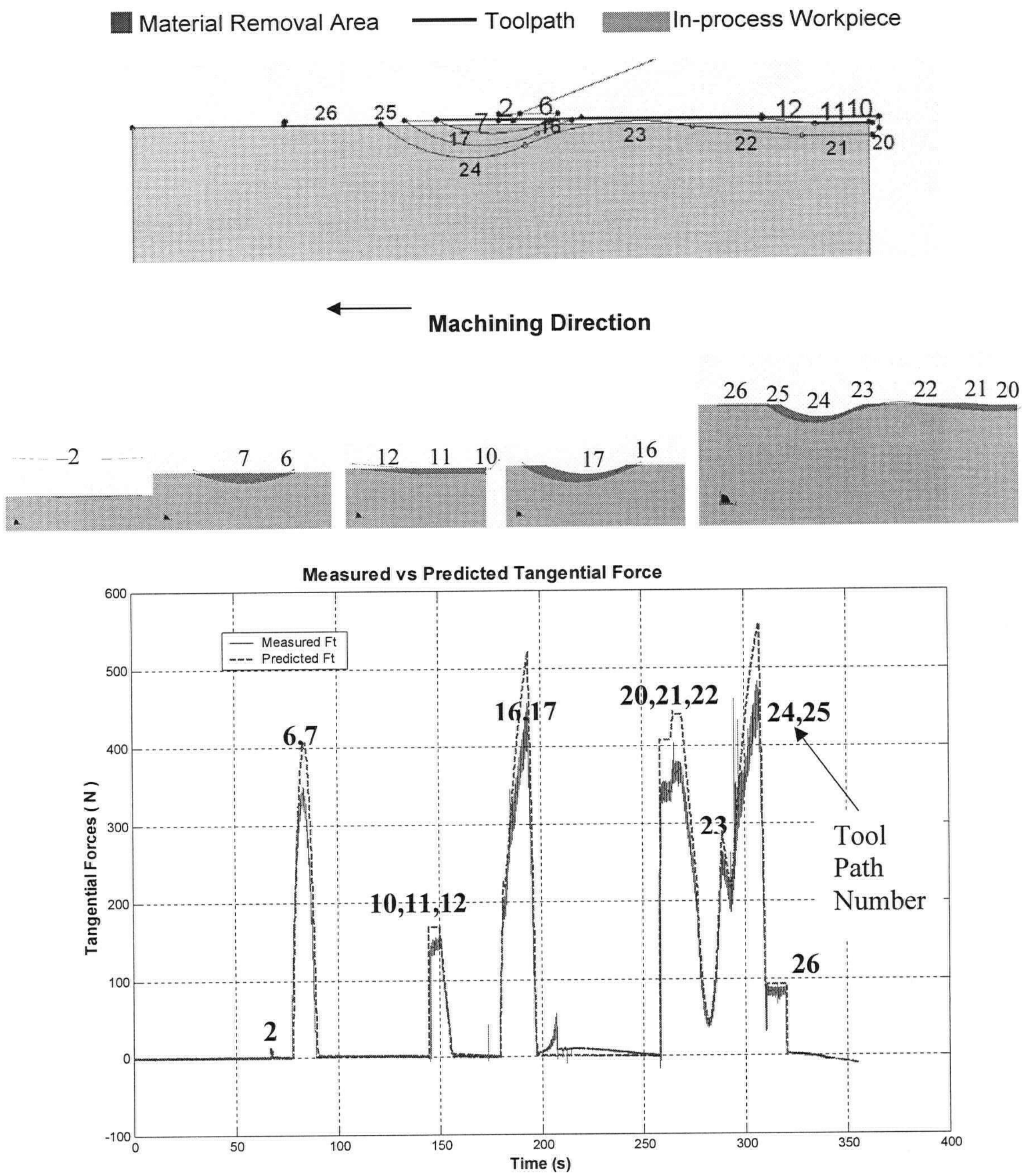


Figure 5.12: Comparisons of the Tangential Forces of First Cut

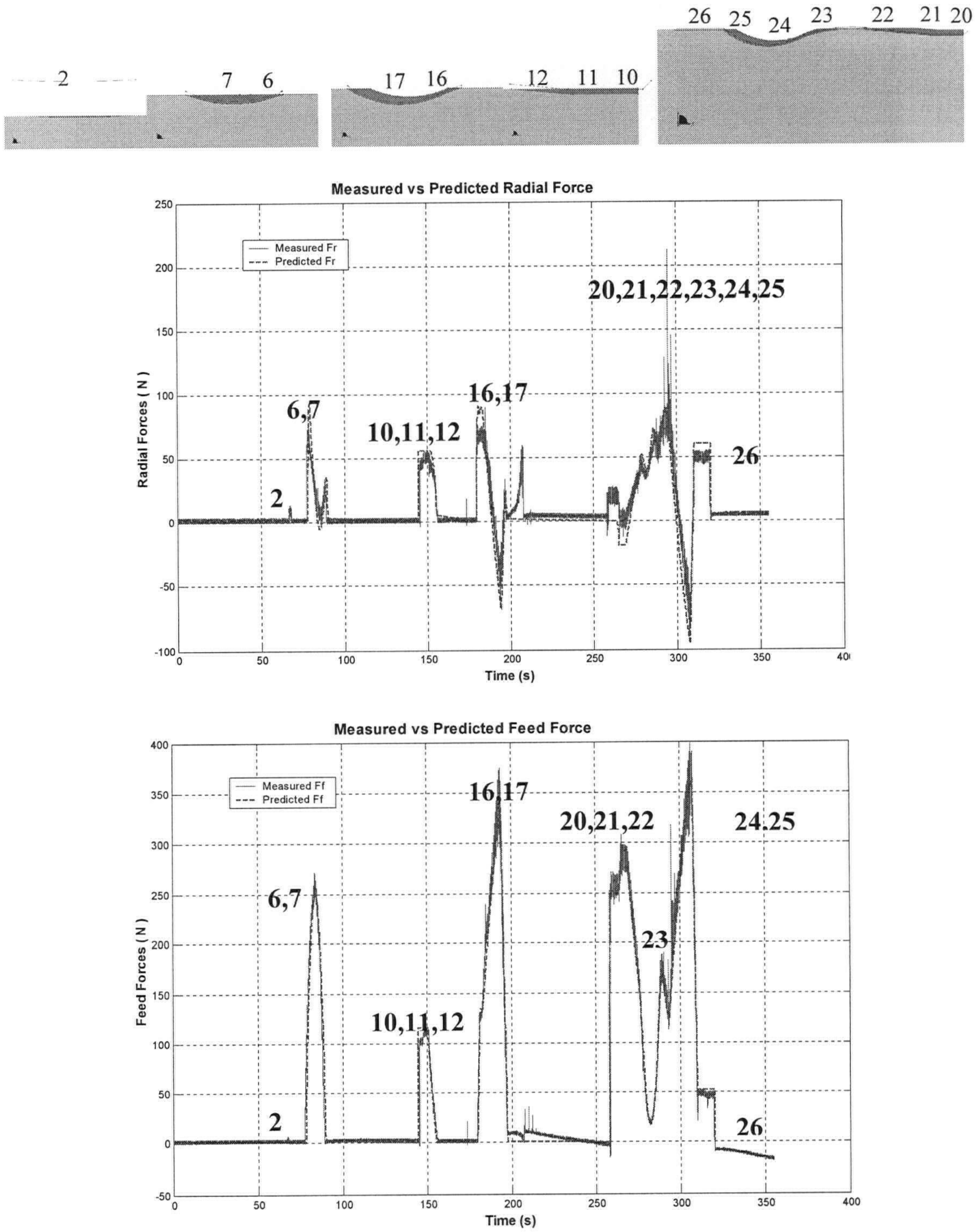


Figure 5.13: Comparisons of the Radial and Feed Forces of First Cut

It must be noted that the unit of the X axis is the machining time as shown in Figures 5.12 and 5.13. Since the machining direction in turning simulation is leftwards, the forces shown in the figures are left and right reversed with respect to the real machining direction.

From the force comparison shown in the above figures, the measured forces and the predicted force are in good agreement. Tangential forces and feed forces are proportional to the tool-workpiece intersection area; i.e., the forces are big when the uncut chip areas are big. This trend is consistent with the adopted mechanistic force model, and demonstrates that the intersection model developed in this research provides correct geometric information along the contour toolpaths. The radial force is not always proportional to the uncut chip area and changes the direction. The reason is that the sign of the radial force in the two regions (tool nose region and straight cutting edge region) is opposite if the side cutting edge angle is positive.

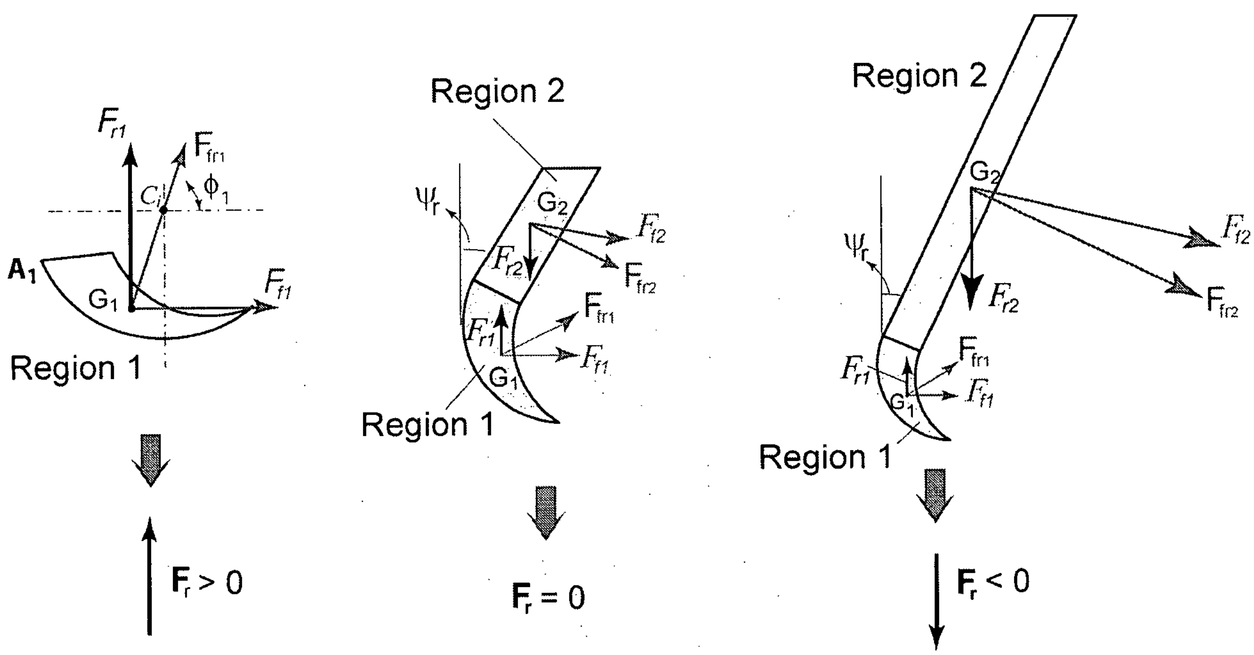
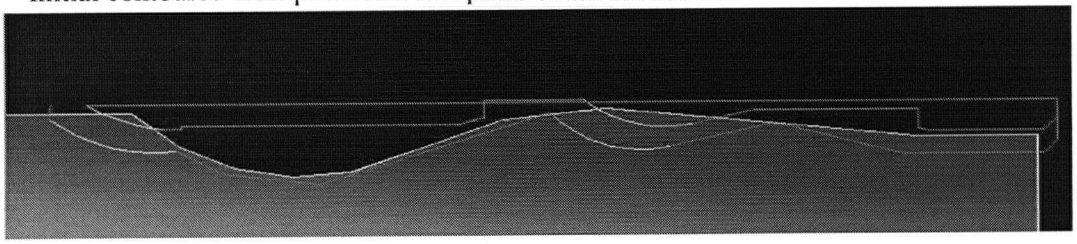


Figure 5.14: The Changes of the Radial Forces with the Depth of Cut

Illustrated in Figure 5.14, the final radial force ($\mathbf{F_r}$) is positive when the uncut chip area only has the tool nose region corresponding to the small depth of cut. When the depth of cut increases, the radial forces in the two regions cancel each other, and $\mathbf{F_r}$ drops to zero in certain depth of cut (shown as the middle case in the above figure). And then $\mathbf{F_r}$ becomes negative and the magnitude increases as the depth of cut increases. This theoretical force analysis is verified by the measured radial data (Figure 5.13). As the result, the predicted radial forces match the measured radial forces very well. The intersections and forces are shown to be accurate and correct in the first cut.

5.5.3 Second Operation

Initial contoured workpiece and tool paths of the second cut



Updated in-process workpiece after subtracting intersections of the second cut

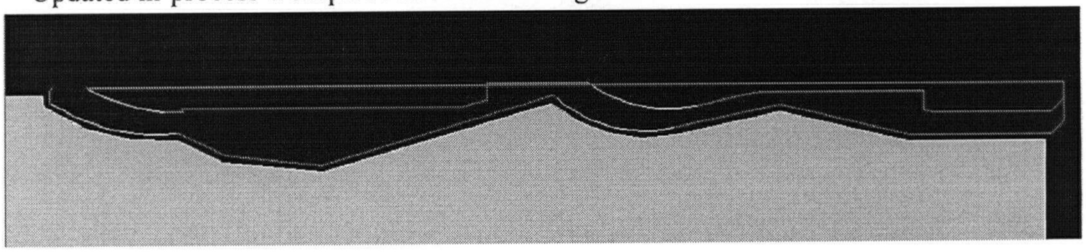


Figure 5.15: Tool Paths and Workpiece of the Second Cut

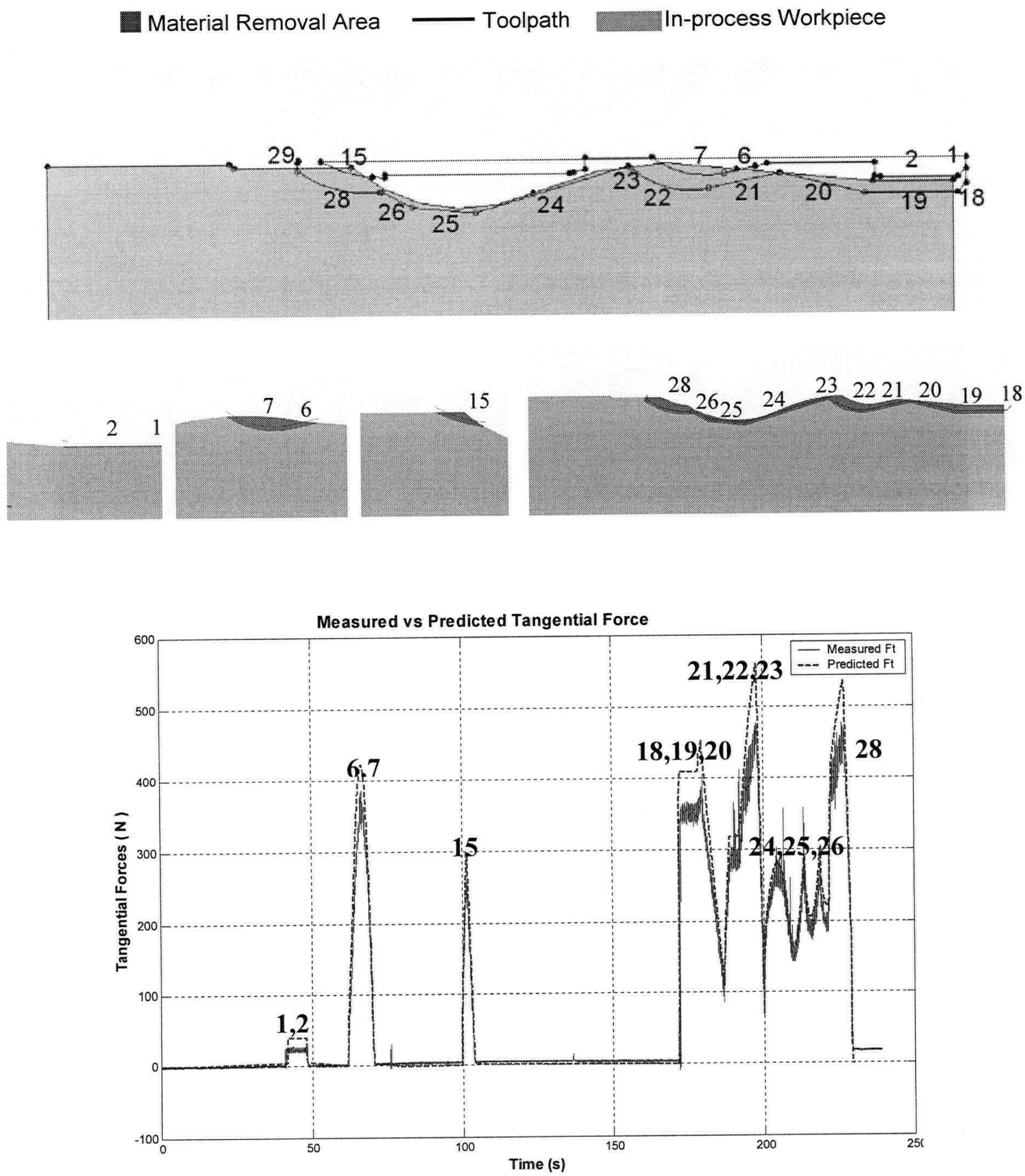


Figure 5.16: Comparisons of the Tangential Forces of the Second Cut

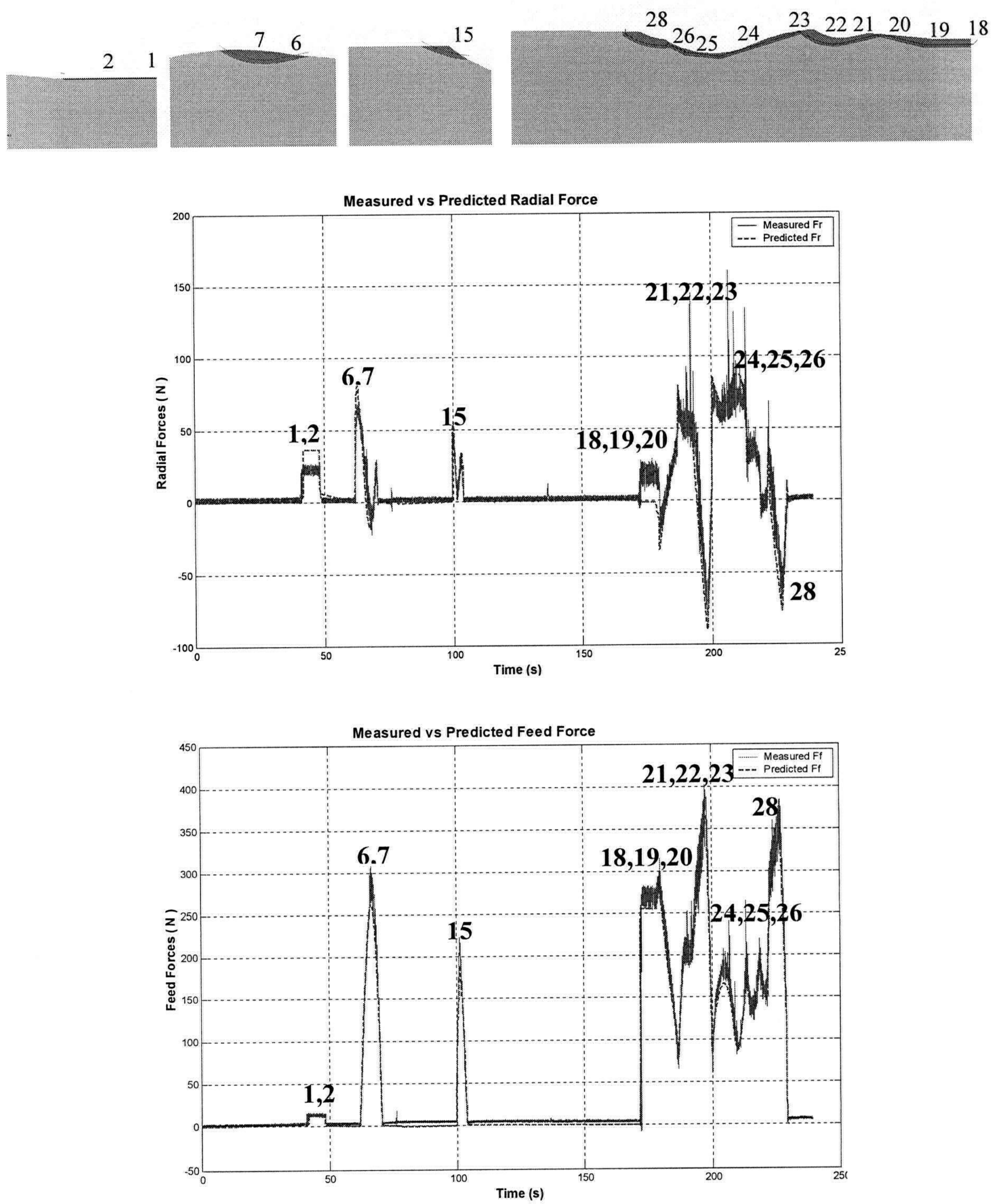
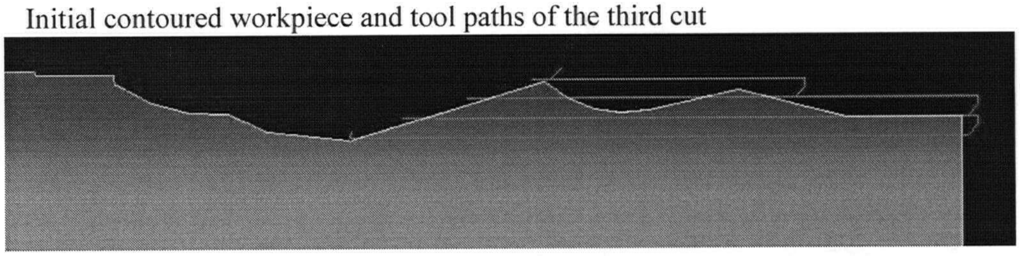


Figure 5.17: Comparisons of the Radial and Feed Forces of the Second Cut

The comparisons of the second cut also present that the predicted forces match the measured forces very well. The good agreement again demonstrates that the proposed mechanistic force model and the intersection methodology are capable of contour turning, in which the workpiece has waved surface and the toolpaths are non-parallel with the workpiece. The challenge here is that the uncut chip area is hard to predict since the workpiece is not simple cylindrical block. By using the proposed hybrid solid modeler, analytical and feature-based method, the geometric information of the tool-workpiece intersections is obtained correctly and effectively. These correct intersections lead to correct predictions of the cutting forces.

5.5.4 Third Operation

Initial contoured workpiece and tool paths of the third cut



Finished final part after subtracting the intersections of the third cut

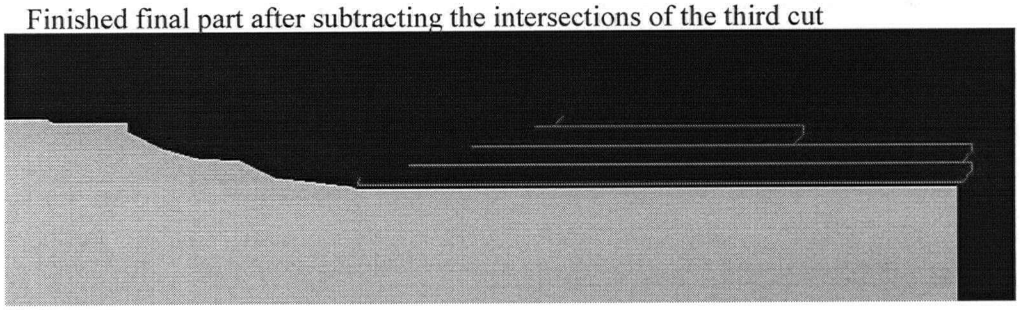


Figure 5.18: Tool Paths and Workpiece of the Third Cut

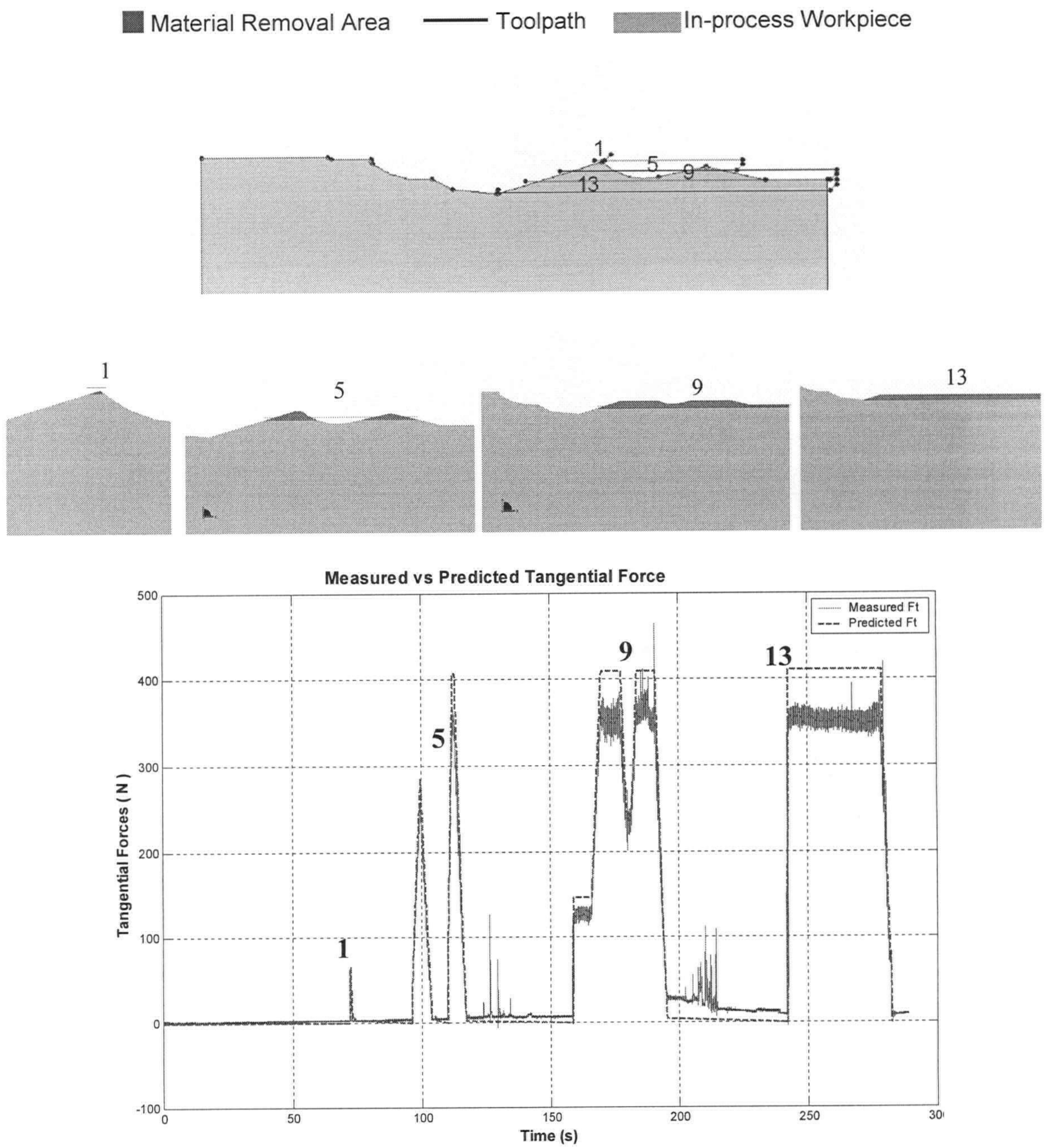


Figure 5.19: Comparisons of the Tangential Forces of the Third Cut

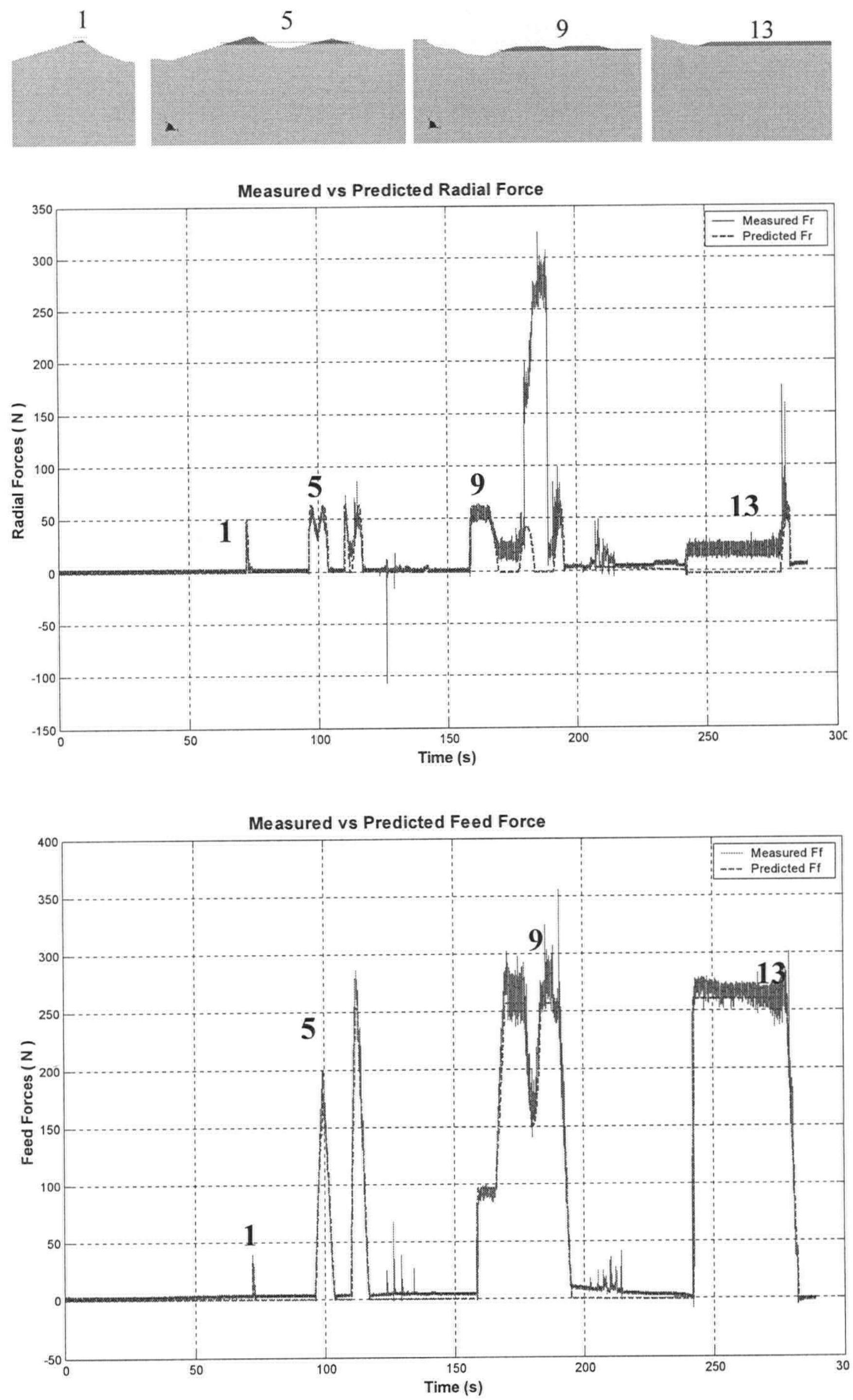


Figure 5.20: Comparisons of the Radial and Feed Forces of the Third Cut

As with the previous two cuts, the predicted forces match well with the measured forces in the third cut. This consistency verifies that the proposed force model and the intersection model are valid for straight toolpaths with contour workpiece.

However, the predicted tangential forces are a little bigger than the measured tangential forces, but the errors remain less than 15%. This deviation may come from the predicted cutting coefficients or from the noise of experiments. There are big discrepancies in the radial forces of Figure 5.20, this may be because the radial forces are very small, less than 50 N. Due to the effects of chips and noise, the measured forces are not completely reliable.

5.6 Conclusion and Future Work

From the force comparisons it can be seen that the measured forces and predicted forces are in good agreement, especially the magnitude and direction of the radial forces. The predicted tangential force is a little bigger than the measured force, but the error is less than 15%.

The conclusions can be summarized as follows.

- The tool-workpiece intersection methodology works well and captures the correct instantaneous uncut chip areas and in-process geometries for force prediction.
- The mechanistic cutting coefficients evaluated from the orthogonal database are accurate enough.
- The assumption that the changes of mechanistic cutting coefficients due to the changes in side cutting edge angle along the contour toolpath (Section 5.4) are neglectable, is acceptable because the forces accurately predicted.

- The mechanistic force model accurately predicts of the instantaneous cutting forces along the contour turning.

To further verify the mechanistic force model, more experiments should be conducted to fully identify the following factors.

- The exact mechanistic cutting coefficients need to be identified from the cutting tests.
- The effect of a change in the side cutting edge angle along the contour toolpath to the mechanistic cutting coefficients, and also the effect on the predicted cutting forces, needs to be investigated.
- More cutting tests on different materials and with different tools.

Overall, the proposed two main models of the Virtual Turning system are verified from the experimental results. The forces at each machining step along the arbitrary contoured toolpath and workpiece are predicted. Later the forces will be used to optimize the turning process by changing the feedrate and cutting speed. This force model can also be easily extended to dynamics of turning.

Chapter 6

Conclusions

6.1 Conclusions

A prototype Virtual Turning system, which can predict the cutting forces, torque, power and deflections along the toolpath, is developed in this thesis.

The system has two integrated components: Tool-workpiece engagement identification based on CAD techniques, and process simulation based on the laws of metal cutting mechanics.

The tool-workpiece intersection is identified from tool geometry, imported workpiece geometry and tool motion information from standard CAD/CAM software systems. Two fundamental approaches are developed to identify the tool engagement conditions. The first method is based on Boolean intersection of tool and workpiece by using their Boundary Representation models in ACIS solid modeling kernel. Since the computational cost is quite high with the first method, a hybrid analytical, feature-based solid modeling approach is developed as a viable alternative. The engagement conditions are grouped as a class of geometric features, and as they are encountered along the toolpath, they are retrieved as opposed to repetitive computation of recurring engagement conditions. Green's theorem is then used to evaluate the chip area at each tool engagement feature. The hybrid model improved the computational efficiency of tool-workpiece intersection by significantly reducing Boolean operations and numerical area calculations in solid modeler.

The process is simulated by using the tool-workpiece intersection and previously developed Mechanistic Model of the turning process. The transformation of orthogonal cutting to discrete, oblique cutting edge elements along the tool engagement zone takes significant computational time which hinders the practicality of Virtual Turning Simulation system. In order to reduce the computational complexity and time, the cutting coefficients are evaluated from the orthogonal cutting database by considering the classified chip features and areas. As a result, the cutting force is predicted as just function of total chip area and cutting edge engagement length estimated from the tool-workpiece intersection engine.

The overall prototype Virtual Turning system is experimentally validated in machining a sample Aluminum workpiece on a CNC lathe. The predicted and measured cutting forces are shown to have sufficient agreement for practical use of the system in basic turning operations.

The contributions of the thesis can be summarized as follows:

- An experimentally validated, prototype Virtual Turning Process simulation system is developed. The system is one of the first reported in the literature.
- A solid modeler-based tool-workpiece intersection algorithm is developed by applying the Boolean intersections of their boundary representation models at each toolpath. The proposed modeling approach reduces the computational cost by using Toolpath Swept Area intersecting with the workpiece at each toolpath in comparison to the tool intersecting with the workpiece at each feed increment. The reduced solid model complexity and number of Boolean operations decreases the computational cost significantly, since the number of toolpaths are used as opposed to the number of feedrate increments which are typically an order of magnitude bigger.
- A Tool Swept Area (*TSA*) algorithm is developed for toolpaths containing line and arc segments based on the tool geometry and a toolpath. The *TSA* is generated by

identifying envelope edges of the path and connecting them with the tool edges. A general convex hull algorithm is used for the linear toolpath *TSA* construction and identified critical points of the tool swept envelopes used for circular toolpath *TSA* construction. The proposed simple algorithm is used to represent tool swept area in two dimensional turning paths in a computationally efficient manner, although it is not applicable to more generic turning operations.

- A hybrid algorithm to evaluate the tool engagement and chip area is developed based on the combination of the solid modeling method, a feature identification algorithm, and an analytical Green's Theorem based method for calculating chip areas . The use of features and analytical formulations for the majority of toolpaths during machining to extract the engagement parameters increases the computational efficiency. The solid modeler is used to construct the workpiece, the tool, and the toolpath, generate material removal areas, and extract tool–workpiece engagements when situations which cannot be handled analytically are encountered. In short, the proposed hybrid technique can handle a variety of cutting tool engagement conditions.
- Tool Workpiece Engagement (*TWE)* geometry has been grouped into a small set of classes the areas and centroids of which are expressed in appropriate formulations that can be solved analytically using Green's Theorem. This method increases the computational efficiency due to two reasons: first, generic numerical solvers in the solid modeler are not used; second, Boolean intersections are not required to obtain the

intersection solid for extracting the required parameters, which is computationally expensive.

- A novel use for in-process machining features has been developed along with feature recognition algorithms. These features are classified as geometric invariant, form invariant and transient features. An area decomposition algorithm is applied to the material removal area along a toolpath segment to generate these features. For a geometric invariant feature, all *TWE* geometry within the feature is the same, thus the boundary position calculations are performed only once. For form invariant features parametric expressions of lines and arcs are used in finding the intersections between linear and circular components. As a result all the boundary positions of a *TWE* are calculated as a function of machining parameters (feedrate, depth of cut), tool geometry (ψ_r, κ_r, r_ε), and tool center positions along the toolpath. In short, with the exception of the transient features, the boundary conditions at each step along the toolpath are determined analytically enhancing computational efficiency.
- A mechanistic force model previously developed at UBC is adopted with slight modifications for improved computational efficiency and force prediction in radial direction. An algorithm is developed to predict cutting forces in contour turning operations, where the tool engagement conditions and the directions of the cutting forces continuously vary. The predicted forces at each feed step are projected to the global XYZ directions of the toolpath, and used in evaluating power, torque, and deflection in contour turning operations.

6.2 Future Research Directions

The proposed Virtual Turning system does not consider the structural dynamics of the system, hence the forced and chatter vibrations are not included in simulations. An accurate prediction of chatter stability and dimensional form errors left on the finish surface are still unresolved research topics, and need to be further investigated before including them in Virtual Turning Simulations.

The thesis dealt only with two dimensional tool-workpiece intersections. In order handle a variety of turning operations, three dimensional workpiece and multi-axis tool motions need to be studied. Parts having slots, holes and other non-symmetric features require three dimensional modeling of tool-part intersection algorithms in order to simulate their turning process in virtual environment.

Bibliography

[1] Atabey, F., "*Modeling of Mechanics and Dynamics of Boring*", *M.A.Sc. Thesis*, University of British Columbia, 2001

[2] Atabey, F., Lazoglu, I., Altintas, Y., "Mechanics of Boring Process Part I", *International Journal of Machine Tools and Manufacture, Design, Research and Application* Vol. 43,Issue 5, pp. 463-476, 2003

[3] Atabey, F., Lazoglu, I., Altintas, Y., "Mechanics of boring processes - Part II: Multi-insert boring heads", *International Journal of Machine Tools and Manufacture, Design, Research and Application* 43 (2003) 477 - 484

[4] Altintas, Y., "*Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design*", Cambridge University Press, 2000.

[5] Altintas, Y.,Spence, A.D., "End milling force algorithms for CAD systems", *Manufacturing Technology CIRP Annals*, Vol. 40, pp. 31-34, 1991.

[6] Altintas, Y.,Spence, A.D., "A Solid Modeller based milling process simulation and planning system", *Transactions of the ASME, Journal of Engineering for Industry*, Vol. 116, pp. 61-69, 1994.

[7] Armarego, E.J.A. and Uthaichaya, M., "Mechanics of Cutting Approach for Force Prediction in Turning Operations", *J. of Engineering Production*, Vol. 1, pp. 118, 1977.

[8] Armarego, E.J.A., Whitfield, R.C., 1985, "Computer Based Modelling of Popular Machining Operations for Force and Power Predictions", *Annals of CIRP*, Vol. 34, pp.65-69, 1985.

[9] Yip-Hoi, D., Dutta, D., Huang, Z., "A Customizable Machining Feature Extraction Methodology for Turned Components", *Journal of Manufacturing Systems*, Vol.22 / No.2, 2003

[10] Yip-Hoi, D., Huang, X., "Cutter Engagement Feature Extraction from Solid Models for End Milling", *Computer Aided Design*, 2004

[11] Carlsson, T., Stiernstoft, T., "A Model for Calculation of the Geometrical Shape of the Cutting Tool-Workpiece Interface", *Annals of the CIRP* Vol.50/1/2001, 41-44

[12] Floriani, L., "Feature Extraction from Boundary Models of Three-Dimensional Objects", *IEEE Transactions on Pattern and Machine Intelligence*. Vol. 11, No.8, August 1989

[13] Fussell, B.K., Hemmett, J.G.,Jerard, R.B. "Geometric and mechanistic modeling integration for five-axis milling force prediction", *Proceedings of the Japan-USA Symposium on Flexible Automation*. Kobe, Japan. Vol. 2. pp. 747-750, 1994.

[14] Fussell, B.K., Jerard, R.B.,Hemmett, J.G., "Modeling of cutting geometry and forces for 5-axis sculptured surface machining", *Computer Aided Design*, Vol. 35, pp. 333-346, 2003.

[15] Fussell, B. K., Jerard, R. B., and Hemmett, J. G., "Robust Feedrate Selection for 3-Axis NC Machining Using Discrete Models", *Journal of Manufacturing Science and Engineering* -- May 2001 -- Volume 123, Issue 2, pp. 214-224

[16] Jerard, R.B., Fussell, B.K., Erean, M.T., and HemmeR, J.G., "Integration of Geometric and Mechanistic Models of NC Machining into an Open-Architecture Machine Tool Controller", *Proc. IMECE, Symp. on Dynamics and Control of Material Removal Processes*, DSC-Vol. 2, (Nov. 2000), ASME, 675-682.

[17] Jimenez, P., Torras, C., "An Orientation-Based Pruning Tool to Speed Up Contact Determination between Translating Polyhedral Models", *International Journal of Robotics Research* Vol. 20, No.6, June 2001, 466-483

[18] Abdel-Malek, K., Yeh, Harn-Jou, "Geometric Representation of the Swept Volume Using Jacobian Rank-deficiency Conditions", *Computer-Aided Design*, Vol 29, No.6, 457-468, 1997

[19] Kim, YJ., Varadhan, G., Lin, M., Manocha, D., "Fast swept volume approximation of complex polyhedral models", *Computer-Aided Design* 36 (2004) 1013-10287.

[20] Lee, J.Y., Kim, K., "A feature-based approach to extracting machining features", *Computer-Aided Design*, Vol.30, No.13, 1019-1035, 1998.

[21] Lee, J., Sung J., Kim, M., "Polygonal boundary approximation for a 2D general sweep based on envelope and Boolean operations" *The Visual Computer* (2000) 16:208–240 c, 2000

[22] Ling, ZK and Chase, T 1996, "Generating the swept area of a body undergoing planar motion," *ASME J. Mech.Design*, Vol 118, pp221-233.

[23] Lazoglu, I., and Liang, S. Y., 1996, "Feedrate Optimization on Complex Workpieces for CNC Milling Machines," *ASME 1996 IMECE Symposium on the Physics of Machining Process-III*, Atlanta GA, pp. 129–138.

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

[25] Montgomery, D. and Altintas, Y., "Mechanism of Cutting Force and Surface Generation in Dynamic Milling", *Transaction of ASME, Journal of Engineering for Industry*, pp. 160-168, vol. 113, 1991

[26] Ozdoganlar, B., Endres, W., "An Analytical Representation of Chip Area for Corner-Radiused Tools Under Both Depth-of-Cut and Feed Variations", *Journal of Manufacturing Science and Engineering,* Vol. 122, Nov. 2000.

[27] Ozdoganlar, O. B., Endres, W. J., 1998, "An Analytical Stability Solution for the Turning Process with Depth-Direction Dynamics and Corner-Radiused Tooliling", *Proceedings of ASME Dynamic Systems and Control Division*, Vol. 64, 511 - 518

[28] Rao, P.N., Rao,J.S., "Towards improved Design of Boring Bars Part 1:Dynamic Cutting Force Model with Continuous System Analysis for the Boring Bar Performance", *Int. J. Mach. Tools Manufacture.*, Vol. 28, No. 1, pp.33-44, 1988.

[29] Reddy, R.G., DeVor, R.E., Kapoor, S., "A mechanistic force model for combined axial–radial contour turning", *International Journal of Machine Tools & Manufacture* 41 (2001) 1551–1572

[30] Reddy, R.G., Kappor, S., Devor, R.E., "A Mechanistic Force Model for Contour Turning", *Journal of Manufacturing Science and Engineering,* August2000, Vol.122, 398-405.

[31] Spence, A.D., Li, Z. "Parallel processing for 2-1/2 D machining simulation", Proceedings of the 6th ACM Symposium on Solid Modeling and Applications. *Ann Arbor, MI.* Vol.pp. 140-148, 2001.

[32] Samaranayake, P. Armarego, E.J.A. "Technological Performance Prediction Models For Turning with Rounded Corner Tools. I - Theoretical Development", *Machining Science and Technology*, Vol. 2, December, 1999.

[33] Subrahmanyam, S., Wozny, M., "An overview of automatic feature recognition techniques for computer-aided process planning", *computers in Industry* 26 (1995) 1-21

[34] Saturley, P.V., Spence, A. D., "Integration of Milling Process Simulation with On-Line Monitoring and Control", *Int J Adv Manuf Technol* (2000) 16:92–99

[35] Takata, S., "A Cutting Simulation System for Machinability Evaluation Using a Workpiece Model", *CIRP Annals*, Vol. 38/1, pp. 417-420, 1989

[36] Tseng, Y.-J, Joshi, S.B., "Recognition of interacting rotational and prismatic machining features form 3D mill-turn parts", *INT. J. PROD. RES.*, 1998, Vol. 36, No.11, 3147-3165

[37] Yang, Z., Abdel-Malek, K., "Approximate swept volumes of NURBS surfaces or solids", *Computer Aided Geometric Design* 22 (2005) 1-26.

Appendix A

Circular Toolpath Tool Swept Area Construction

A.1 Critical Position Calculations

To construct the *TSA* of the circular toolpath, the outer and inner sweep envelope edges need to be generated. The corresponding critical points P_1 to P_9 need to be calculated.

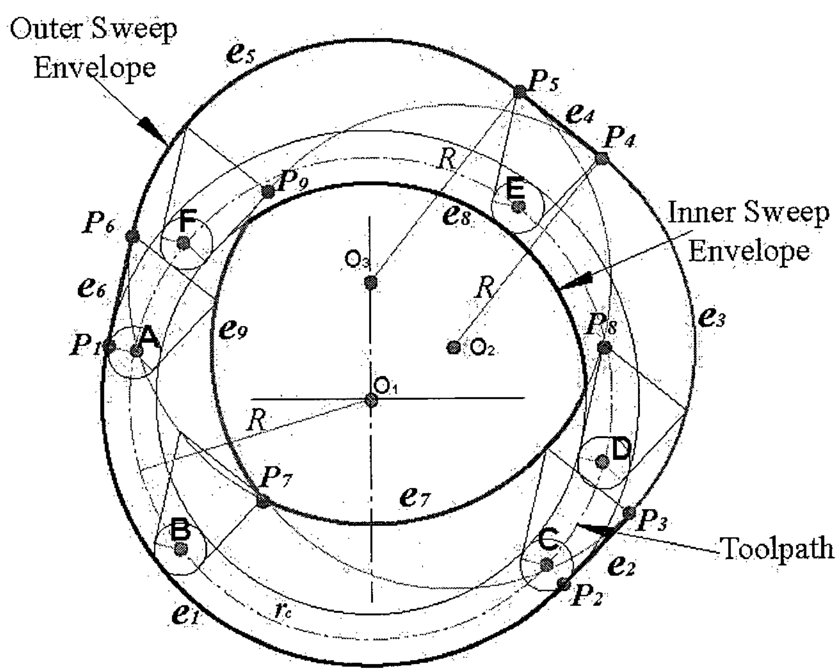


Figure A.1 Circular Toolpath Tool Swept Area Construction

As illustrated in Figure A.1, the Swept Area is the region between the Outer Sweep Envelope and the Inner Sweep Envelope. The Outer Sweep Envelope consists of three circular edges e_1, e_3, e_5 and three straight edges e_2, e_4, e_6. The first three edges are generated from the portion of the sweeping envelope of the tool nose curve, tool upper right point P_c, and tool upper left point P_d (See Figure 3.6). The centers of these three arcs are O_1, O_2 and O_3, and the toolpath radius is R. The latter three edges come from the tool boundary edges that are tangent to the first three circular edges respectively. Similarly, the Inner Sweep Envelope consists of three circular edges e_7, e_8, and e_9, which are generated from another portion of the sweeping envelope of the tool upper left point P_d, the tool nose curve, and the tool upper right point P_c.

Given the tool geometry (r_ε, ψ_r, κ_r), the toolpath geometry (R, O_1), and according to the properties of common tangent and planar rigid motion, the following formulas can be established:

O2
$$\begin{cases} X_{O_2} = X_{O_1} + l_1 \cos(\kappa_r) + r_\varepsilon \sin(\kappa_r) \\ Y_{O_2} = Y_{O_1} + l_1 \sin(\kappa_r) - r_\varepsilon \cos(\kappa_r) \end{cases} \tag{A.1}$$

O3
$$\begin{cases} X_{O_3} = X_{O_1} - r_\varepsilon \cdot \cos(\psi_r) + (L - r_\varepsilon(1 + \sin(\psi_r)))\tan(\psi_r) \\ Y_{O_3} = Y_{O_1} + L - r_\varepsilon \end{cases} \tag{A.2}$$

V1
$$O_3 - O_2 \Big/ |O_3 O_2| = (v_{1x},\ v_{1y}) \tag{A.3}$$

P1:
$$\begin{aligned} X_1 &= X_{O_1} - (R + r_\varepsilon) \cdot \cos(\psi_r) \\ Y_1 &= Y_{O_1} + (R + r_\varepsilon) \cdot \sin(\psi_r) \end{aligned} \tag{A.4}$$

P2:
$$\begin{aligned} X_2 &= X_{O_1} + (R + r_\varepsilon) \cdot \sin(\kappa_r) \\ Y_2 &= Y_{O_1} - (R + r_\varepsilon) \cdot \cos(\kappa_r) \end{aligned} \tag{A.5}$$

P3:
$$\begin{aligned} X_3 &= X_{O_1} + l_1 \cdot \cos(\kappa_r) + (r_\varepsilon + R) \cdot \sin(\kappa_r) \\ Y_3 &= Y_{O_1} + l_1 \cdot \sin(\psi_r) - (r_\varepsilon + R) \cdot \cos(\kappa_r) \\ &\text{where } l_1 = \frac{(W - r_\varepsilon(1 + \sin\psi_r))}{\cos(\psi_r + \kappa_r)} \end{aligned} \tag{A.6}$$

P4:
$$\begin{aligned} X_4 &= X_{O_1} + (R + l_1) \cdot \cos(\kappa_r) + r_\varepsilon \cdot \sin(\kappa_r) \\ Y_4 &= Y_{O_1} + (R + l_1) \cdot \sin(\psi_r) - r_\varepsilon \cdot \cos(\kappa_r) \end{aligned} \tag{A.7}$$

P5:
$$X_5 = X_{O_1} - r_\varepsilon \cdot \cos\psi_r + R \cdot \cos\kappa_r + (L - r_\varepsilon(1+\sin\psi_r)) \cdot \tan\psi_r$$
$$Y_5 = Y_{O_1} + L - r_\varepsilon + R \cdot \sin(\kappa_r) \tag{A.8}$$

P6:
$$X_6 = X_{O_1} - (r_\varepsilon + R) \cdot \cos\psi_r + (L - r_\varepsilon(1+\sin\psi)) \cdot \tan\psi_r$$
$$Y_6 = Y_{O_1} + L - r_\varepsilon + R \cdot \sin(\psi_r) \tag{A.9}$$

P7:
$$\frac{O_2 + O_3}{2} + \vec{v}_2 \cdot d,$$
$$\text{where } \vec{v}_2 = (-v_{1y}, \quad v_{1x}) \tag{A.10}$$
$$d = \sqrt{R_2 - \left(\left|O_2O_3\right|\big/2\right)^2}$$

P8:
$$X_8 = X_{O_3} + R \cdot \cos(\psi_r)$$
$$Y_8 = Y_{O_3} - R \cdot \sin(\psi_r) \tag{A.11}$$

P9:
$$X_9 = X_{O_2} + R \cdot \sin(\psi_r)$$
$$Y_9 = Y_{O_2} - R \cdot \cos(\psi_r) \tag{A.12}$$

A.2 Tool Swept Area of Partial Circular Tool path

In the real turning operation, it is common that a portion of the circle tool path is encountered. To determine the tool swept area of a given tool path, the location of the given tool path needs to be identified at first, i.e., to find the given tool path belongs to which section or which combined sections with respect to the whole circle toolpath. According to the critical positions P_1 to P_9, and the tool geometry relationship, the corresponding critical toolpath positions A to F are calculated. Therefore, six sections of toolpath are constructed, i.e., AB, BC, CD, DE, EF and FA. Then the tool swept area is constructed by generating the outer and inner edge loops based on the boundary conditions of those sections.

A
$$\begin{aligned} X_A &= X_{O_1} - R \cdot \cos(\psi_r) \\ Y_B &= Y_{O_1} + R \cdot \sin(\psi_r) \end{aligned} \tag{A.13}$$

B
$$\begin{cases} X_B = X_7 + l_1 \cos(\kappa_r) + r_\varepsilon \sin(\kappa_r) \\ Y_B = Y_7 + l_1 \sin(\kappa_r) - r_\varepsilon \cos(\kappa_r) \end{cases} \tag{A.14}$$

C
$$\begin{aligned} X_C &= X_{O_1} + R \cdot \sin(\kappa_r) \\ Y_C &= Y_{O_1} - R \cdot \cos(\kappa_r) \end{aligned} \tag{A.15}$$

D
$$\begin{cases} X_B = X_7 - r_\varepsilon \cdot \cos(\psi_r) + (L - r_\varepsilon(1 + \sin(\psi_r))) \tan(\psi_r) \\ Y_B = Y_7 + L - r_\varepsilon \end{cases} \tag{A.16}$$

E
$$\begin{aligned} X_E &= X_{O_1} + R \cdot \cos(\psi_r) \\ Y_E &= Y_{O_1} - R \cdot \sin(\psi_r) \end{aligned} \tag{A.17}$$

F
$$\begin{aligned} X_F &= X_{O_1} + R \cdot \sin(\psi_r) \\ Y_F &= Y_{O_1} - R \cdot \cos(\psi_r) \end{aligned} \tag{A.18}$$

One example of a circular toolpath TSA construction is presented in the follows.

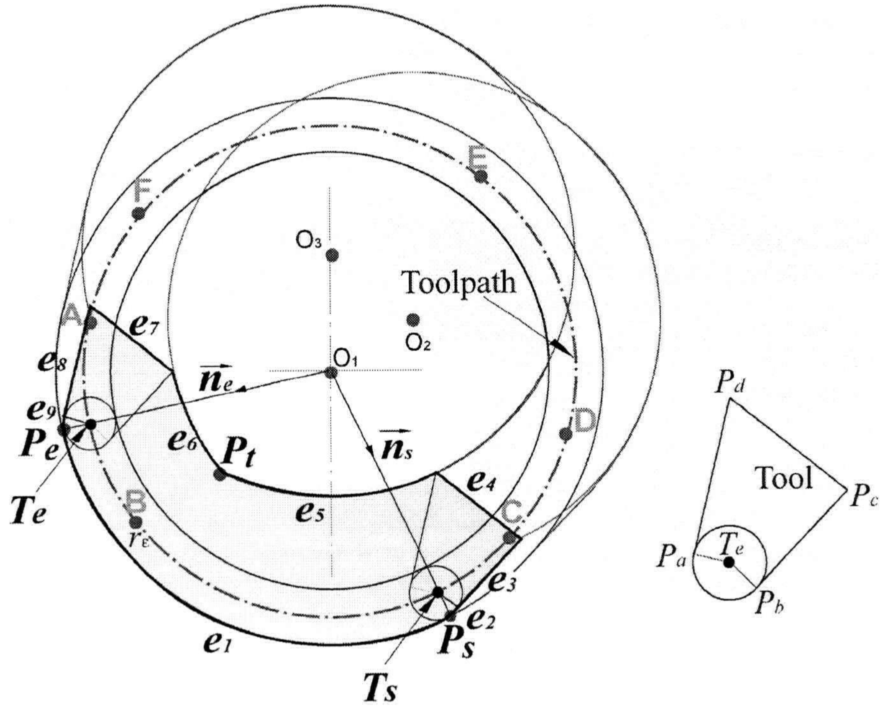


Figure A.2 Tool Swept Area of T_eT_s Construction

As shown in Figure A.2, the given toolpath T_eT_s is inside AB and BC sections after section identifications. It is known that outer envelope edge is e_1 and the inner envelop edge is e_6 and e_5 in section AB and BC respectively. Therefore the swept area is the combination of the envelope edges e_1, e_5 and e_6, and tool boundary edges at toolpath start and end positions. Assume the tool points at Te are $\{P_a, P_b, P_c, P_d\}$, tool points at Ts are $\{P_a', P_b', P_c', P_d'\}$, they can all be calculated by using T_e, T_s, and tool construction equations (see Figure 3.6). Outer envelope e_1 can be constructed as follows:

Unit direction vector $$\vec{n}_e = {T_e - O_1}\Big/{|T_eO_1|} \tag{A.19}$$

e_1 end point P_e: $$P_e = T_e + \vec{n}_e \cdot r_\varepsilon \tag{A.20}$$

Similarly, $$\vec{n}_s = {T_s - O_1}\Big/{|T_sO_1|} \tag{A.21}$$

e_1 start point P_s: $$P_s = T_s + \vec{n}_s \cdot r_\varepsilon \tag{A.22}$$

From equations (A.19 ~ A.22), the end positions of e_1 are calculated, along with $O_1$and R, e_1 is constructed.

As shown in Figure A.2, P_t is the connected point between e_5 and e_6 and calculated in equation (A.10). e_5 is constructed by center O_3, end points P_d' and P_t; e_6 is constructed by center O_2, end points P_t and P_c.

Finally, the *TSA* of T_eT_s is consisted of a list of counter clockwise edges: $\{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\}$. Where e_2, e_3 and e_4 are the tool edges at the start position and e_7, e_8 and e_9 are the tool edges at the end position. e_1 and e_5, e_6 are the outer envelope and inner envelope edges. *TSA* of other sections is constructed in the similar manner.

Appendix B

Green's Theorem-Based Analytical Area Calculation

B.1 Classes of Generic Tool Engagement Features (*teF*)

Figure B.1 illustrates six commonly occurring *teFs* that have been identified for *TWE* calculations.

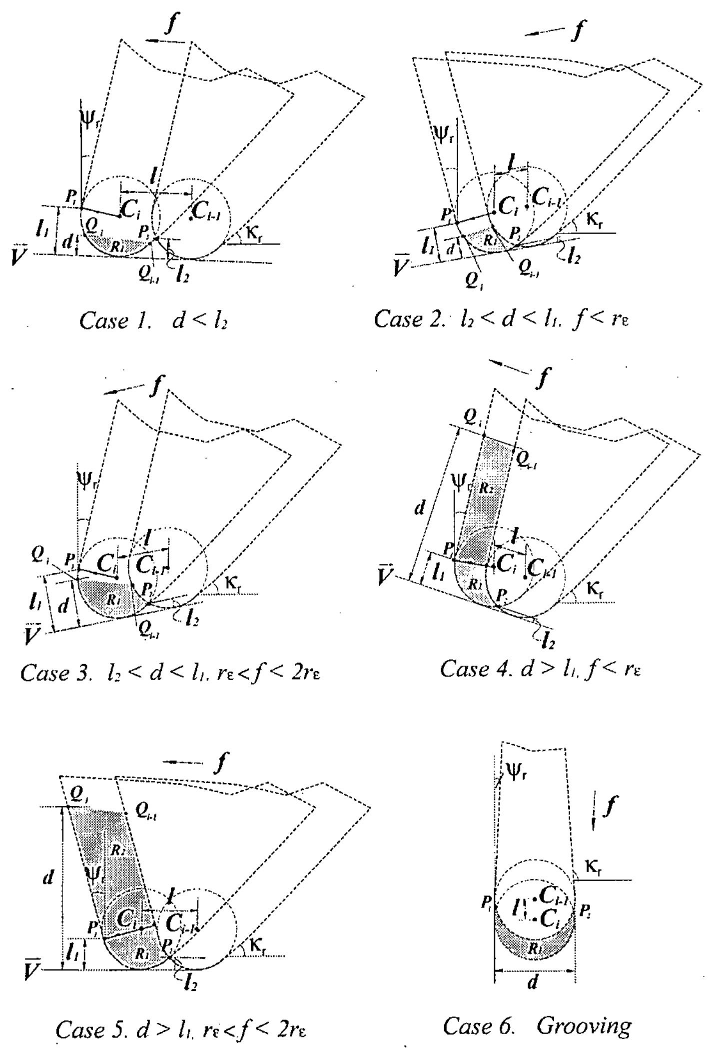


Figure B.1 Classes of Generic Tool Engagement Features (*teF*)

In Figure B.1, l is the distance between the successive two tool positions C_iC_{i-1}. If the tool path is a straight line, l is equal to feedrate f. If the tool path is a circular edge, f is the arc length and l is the chord length. P_1 is the tool nose curve upper tangent point of the tool, and P_2 is the intersection point between the two tools. Note that P_2 can be the intersection between two tool nose curves or between tool nose curve and major straight cutting edge (when f is small) or minor straight cutting edge and tool nose curve (when f is big). Identifying the types of P_2 is included in the detailed algorithm. $\vec{V}$ denotes the instantaneous feed direction that is tangent to the tool nose curve at C_i, and l_1 and l_2 are the distances from P_1, P_2 to $\vec{V}$ respectively. The depth of cut d is the distance from the workpiece boundary to $\vec{V}$. In case 1, the *teF* only has one region R_1 and is covered by two edges. In case 2 and 4, depth of cut d is smaller than l_1 but bigger than l_2, there are only R_1 region in both cases. In case 3 and case 5, d is bigger than l_1, hence there are two regions in each case.

B.2 General Area Calculation Algorithm

After identifying the six types of *teFs*, appropriated analytical equations can be formulated for each case by giving the boundary conditions of the engagement, such as the numbers of enclosed edges, properties of edges (linear or circular). All the intersecting positions and angles, which are required by these formulations and treated as the inputs to get the final results, are also analytically derived from Feature identification algorithm and will be described in Appendix C in detail.

A general Green's Theorem-based area calculation algorithm is expressed in the following algorithm. *teF* of each case presented in Figure B.1 follows this algorithm, and it is integrated and rearranged to form some fixed formulations which will be shown in the next section.

Algorithm **Area_Calculation**

INPUT: $C_i, C_{i-1}, \{P_i\}, e_i, n, m, k, f$

($\{P_i\}$: the set of end points, n: number of edges, m: number of circular edges, k: number of zones)

OUTPUT: A_1, A_2, A

STEP:

For region R_j, $j = 1$ to 2

From $i = 1$ to n

CASE geometry_type (e_i) ≡ LINE

$$A_i = \int_0^1 -yx'\,du = \frac{(y_{i+1} + y_i)(x_{i+1} - x_i)}{2},$$

where $P_{i+1}(x_{i+1}, y_{i+1})$, $P_i(x_i, y_i)$

CASE geometry_type(e_i) ≡ ARC

If center is C_i

$$A_i = \int_{\theta_1}^{\theta_2} -yx'\,du = \frac{r^2 u}{2} - rY_c \cos(u) - \frac{r^2}{4}\sin(2u)\Bigg|_{\theta_1}^{\theta_2},$$

where $\theta_1 = \pi + \arctan(y_{P_iC_i} / x_{PiCi})$ $\qquad 0 \le \theta_1 < 2\pi$

$\theta_2 = 2\pi + \arctan 2(y_{P_{i+1}C_i}, x_{P_{i+1}C_i})$ $\qquad 0 \le \theta_2 < 2\pi$

If center is C_{i-1}

$$A_i = \int_{\psi_1}^{\psi_2} -yx'\,du = \frac{r^2 u}{2} - rY_c \cos(u) - \frac{r^2}{4}\sin(2u)\Bigg|_{\psi_1}^{\psi_2}$$

where $\psi_1 = 2\pi + \arctan 2(y_{P_iC_{i-1}}, x_{P_iC_{i-1}})$ $\qquad 0 \le \psi_1 < 2\pi$

$\psi_2 = \pi + \arctan(y_{P_{i+1}C_{i-1}} / x_{P_{i+1}C_{i-1}})$ $\qquad 0 \le \psi_2 < 2\pi$

$A_j \leftarrow A_i + A_j$

End

End

$A = A_1 + A_2$

End

B.3 Area Calculation Derivation for *teF4*

Detailed derivation of formulations is given to *teF4*, because it is the most commonly encountered engagement type. The final expressions of other types of *teF* are listed in the next section.

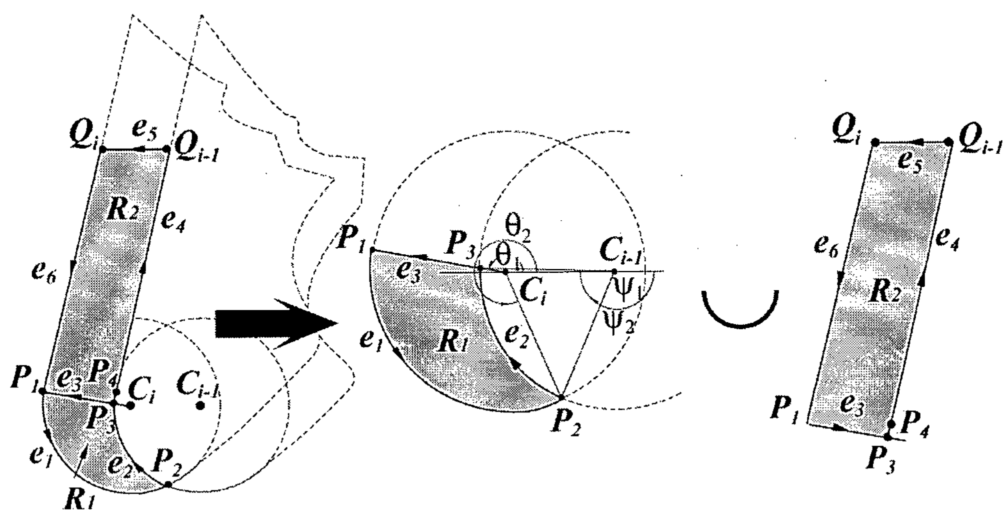


Figure B.2: *teF4* Area Calculation

In B.2, points $C_i(Xc_i, Yc_i)$, $C_{i-1}(Xc_{i-1}, Yc_{i-1})$ are the consequence tool nose center positions along the tool path. The interval is $l = |C_iC_{i-l}|$. *TWE* has two regions R_1, R_2 for cutting force calculation. P_1, P_4 are the tool nose curve upper tangent points at C_i, C_{i-1} . P_2 is the intersection point between C_i and C_{i-1} . Q_i, Q_{i-1} are the workpiece boundary positions intersected with the tools at the two positions. The coordinates of these boundary points are $P_i(x_i, y_i)$, $Q_i(x_5, y_5)$, and $Q_{i-1}(x_4, y_4)$, θ_1 and θ_2 are the angles of vector P_1C_i, P_2C_i, and ψ_1, ψ_2 are the angles of vector P_2C_{i-1}, P_3C_{i-1}. All the boundary conditions (point coordinates and angles) are derivable analytically as shown in Appendix C.

At first, the type of intersection point P_2 needs to be identified, if the absolute value of equivalent side cutting edge angle, $\psi_r' = \psi_r + \alpha$ $(\psi_r' < 0)$, is smaller than a critical angle $|\psi_r'| \leq \cos^{-1}\left(\dfrac{l}{2r_\varepsilon}\right)$, P_2 is an intersection point between two curves, Figure B.2 presents this intersection type. Figure B.3 shows the critical angle $|\psi_r'| = \cos^{-1}\left(\dfrac{l}{2r_\varepsilon}\right)$, where P_2 and P_4 overlap.

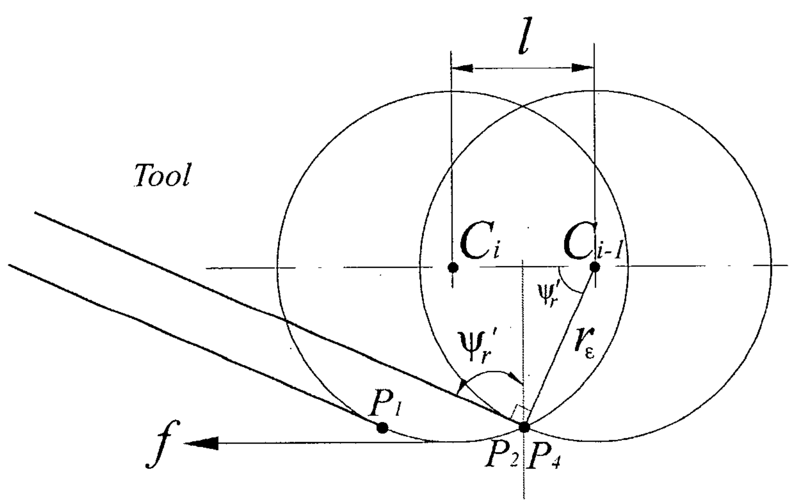


Figure B.3: The Type of Intersection Point P_2

If ψ_r' is positive as shown in Figure B.3, curve region R_1 is known to be bounded by edges $\{e_1, e_2, e_3\}$, and the close to polygonal region R_2 is bounded by edges $\{e_3, e_4, e_5, e_6\}$. e_1, e_2 are a portion of the tool nose curves, and e_3 is the line segment P_1C_i truncated by e_2 at P_3. e_4, e_6 are tool straight cutting edges, and e_5 is a portion of the workpiece boundary edge. The signed areas covered by all the edges are formulated and summed up to give the total area equations.

The edge e_1 is an arc from $P_{1,i}$ to $P_{2,i}$, which is corresponding to the angles θ_1, θ_2, illustrated in Figure B.4.

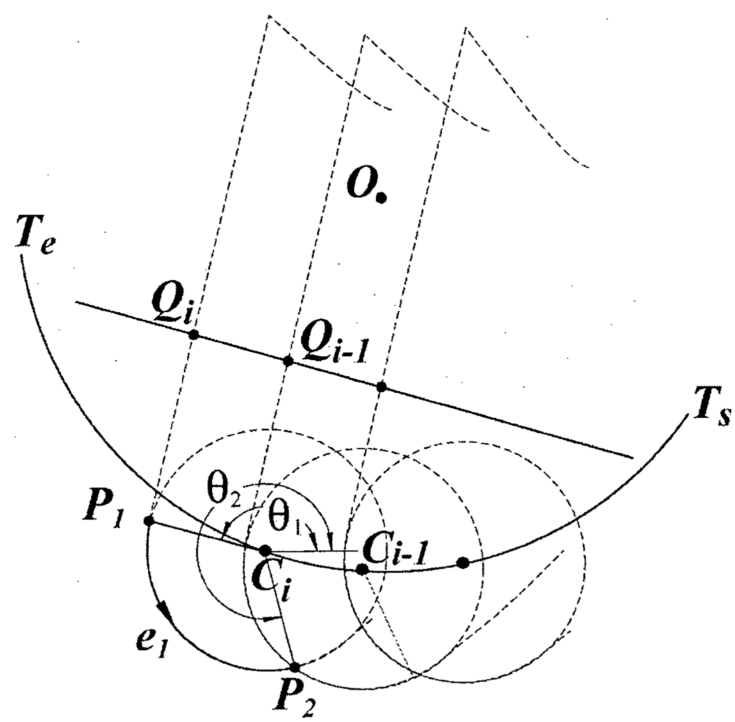


Figure B.4 Area Calculation of Edge e_1

The parametric equation of e_1 is:

$$e_1 : \begin{cases} X = Xc_i + r_\varepsilon \cos(u) \\ Y = Yc_i + r_\varepsilon \sin(u) \end{cases}, \quad \theta_1 \le u \le \theta_2 \tag{B.1}$$

The area A_{11} that is covered by e_1 is:

$$A_{11} = \int_{\theta_1}^{\theta_2} -YX'du = \frac{r_\varepsilon^{\,2} u}{2} - r_\varepsilon Yc_i \cos(u) - \frac{r_\varepsilon^{\,2}}{4}\sin(2u)\Bigg|_{\theta_1}^{\theta_2}$$
$$= \frac{-r_\varepsilon}{4}\left(4Yc_i\left(\cos(\theta_2)-\cos(\theta_1)\right)+r_\varepsilon(\sin(2\theta_2)-\sin(2\theta_1))-2r_\varepsilon(\theta_2-\theta_1)\right) \tag{B.2}$$

The edge e_2 is an arc from $P_{2,i}$ to $P_{3,i}$, which is corresponding to the angles ψ_1, ψ_2, illustrated in Figure B.5.

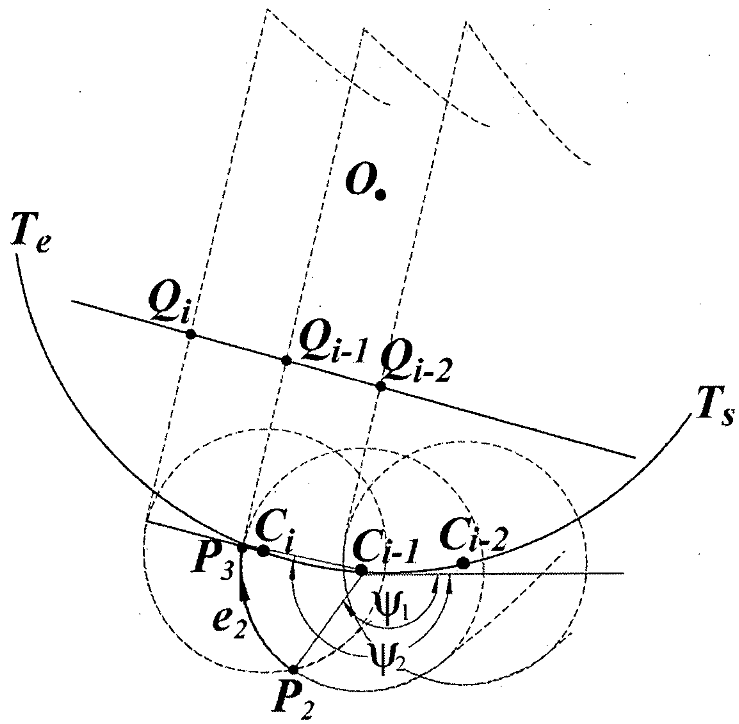


Figure B.5 Area Calculation of Edge e_2

The parametric equation of e_2 is:

$$e_2 : \begin{cases} X = Xc_{i-1} + r_\varepsilon \cos(u) \\ Y = Yc_{i-1} + r_\varepsilon \sin(u) \end{cases}, \quad \psi_1 \le u \le \psi_2 \tag{B.3}$$

The area A_{12} that is covered by e_2 is:

$$
\begin{aligned}
A_{12} &= \int_{\psi_1}^{\psi_2} - yx'\,du = \frac{r_\varepsilon^{\;2} u}{2} - r_\varepsilon Yc_{i-1}\cos(u) - \frac{r_\varepsilon^{\;2}}{4}\sin(2u)\Big|_{\psi_1}^{\psi_2} \\
&= \frac{-r_\varepsilon}{4}\left(4Yc_{i-1}\left(\cos(\psi_2) - \cos(\psi_1)\right) + r_\varepsilon(\sin(2\psi_2) - \sin(2\psi_1)) - 2r_\varepsilon(\psi_2 - \psi_1)\right)
\end{aligned}
\tag{B.4}
$$

The edge e_3 is a line from P_3 to P_1. The parametric equation of e3 is:

$$
e_3 = \begin{cases} X = (1-u)X_3 + uX_1 \\ Y = (1-u)Y_3 + uY_1 \end{cases}, \quad 0 \le u \le 1 \tag{B.5}
$$

The area A_{13} covered by e_3 is:

$$
A_{13} = \int_0^1 - yx'\,du = \frac{(Y_3 + Y_1)(X_3 - X_1)}{2} \tag{B.6}
$$

The total area of region R_1 is:

$$
\begin{aligned}
A_1 &= A_{11} + A_{12} + A_{13} \\
&= \frac{-r_\varepsilon}{4}\big(4\left[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)\right] \\
&\quad + r_\varepsilon\left[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)\right] \\
&\quad - 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1)\big) + \frac{1}{2}(Y_3 + Y_1)(X_3 - X_1)
\end{aligned}
\tag{B.7}
$$

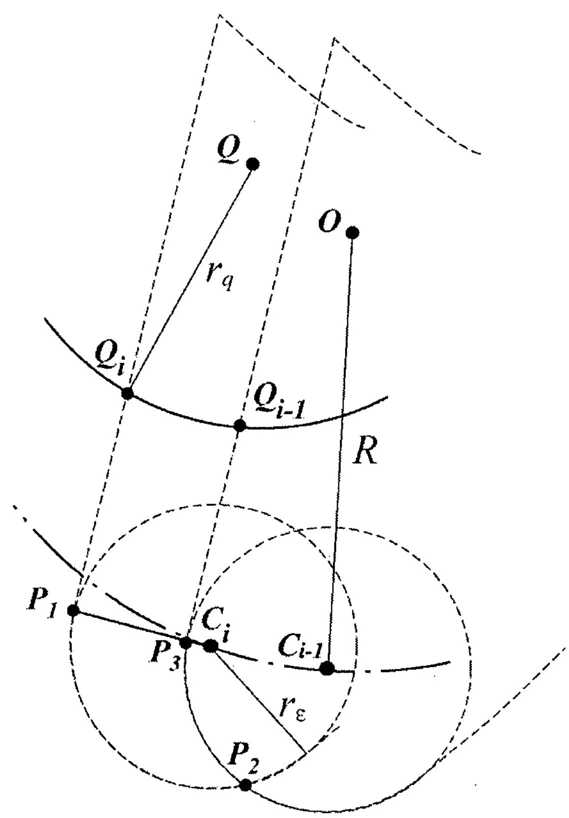
For region R_2, since f is fairly small ($f < r_\varepsilon$), P_3 and P_4 are very close, the small arc segment between them can be approximated as a line segment. The accuracy lost here is neglectable. Therefore, one straight edge e_4 is used to represent the connection between P_3 to Q_{i-1}. As a result, if Q_iQ_{i-1} is a linear component or its radius is fairly big compared to the feedrate, region R_2 is close to a polygonal region, and according to Green's Theorem, the area A_2 is formulated as follows:

$$
\begin{aligned}
A_2 &= \frac{1}{2}\sum_{i=0}^{n-1}(X_iY_{i+1} - X_{i+1}Y_i) \\
&= \frac{1}{2}\left(X_3Y_4 - X_4Y_3 + X_4Y_5 - X_5Y_4 + X_5Y_1 - X_1Y_5 + X_1Y_3 - X_3Y_1\right)
\end{aligned} \tag{B.8}
$$

where $P_1(X_1, Y_1)$, $P_3(X_3, Y_3)$, $Q_{i-1}(X_4, Y_4)$ and $Q_i(X_5, Y_5)$.

B.4 Analytical Area Formulations for *teFs*

The area calculations are derived in the same manner as shown in the previous section. The coordinates of boundary conditions are shown in Figure B.6.



Boundary positions:

Tool centers on the toolpath:
C_{i-1} (Xc_{i-1}, Yc_{i-1})
C_i (Xc_i, Yc_i)

Tool nose radius: r_ε

Intersection points:
P_1 (X_1, Y_1)
P_2 (X_2, Y_2)
P_3 (X_3, Y_3)

Workpiece boundary:
Q_{i-1} (X_4, Y_4)
Q_i (X_5, Y_5)

If $Q_{i-1}Q_i$ is a circular edge:
Center: Q (Xq, Yq), radius: r_q

Figure B.6 Area Calculation of *teFs*

The final results that are used directly in Virtual Machining system are listed in Table B.1. Other engagement characteristics, such as gravity centers, chip-side cutting edge contact length, are pre-formulated in the same method and are used the developed system, they are not listed here due to the space limitation. Also the extreme cases of the type of P_2, i.e., the intersection between tool nose curve and straight cutting edge, is not commonly encountered if f is small, hence, only curve-curve intersection is considered in this table when $f < r_\varepsilon$. As shown in Figure B.5, workpiece boundary segment Q_iQ_{i-1} can be linear or circular edge, and it may intersect with the tool nose edge or the tool side cutting edge.

Table B.1 Green's Theorem-based Area Formulations for all *teFs*

teF1	$A_1 = \frac{1}{2} r_\varepsilon (\phi - \sin(\phi)),\ \phi = 2\cos^{-1}(\frac{r_\varepsilon - d}{r_\varepsilon})$ $A_2 = 0$	(B.9)
teF2	If Q_iQ_{i-1} is a linear component $A_1 = \frac{-r_\varepsilon}{4}(4[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)]$ $+ r_\varepsilon[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)]$ $- 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1)) + \frac{1}{2}(Y_5 + Y_4)(X_5 - X_4)$ If Q_iQ_{i-1} is a circular component	(B.10)

	$A_1 = \frac{-r_\varepsilon}{4}(4[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)] + r_\varepsilon[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)] - 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1)) + \frac{-r_q}{4}(4Y_q(\cos(\varphi_2) - \cos(\varphi_1)) + r_q(\sin(2\varphi_2) - \sin(2\varphi_1)) - 2r_q(\varphi_2 - \varphi_1))$ where θ_1 and θ_2 are the angles of vector Q_iC_i, P_2C_i, ψ_1, ψ_2 are the angles of vector P_2C_{i-1}, $Q_{i-1}C_{i-1}$, and φ_1, φ_2 are the angles of vector $Q_{i-1}Q$, Q_iQ. r_q is the radius of the circular workpiece boundary. $A_2 = 0$	
teF3	If P_2 is the intersection between two curves, same as *teF2* If P_2 is the intersection between tool straight cutting edge and tool nose curve, $A_1 = \frac{-r_\varepsilon}{4}(4[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)] + r_\varepsilon[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)] - 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1)) + \frac{1}{2}(Y_2 + Y_b)(X_2 - X_b) + A_{Q_iQ_{i-1}}$ where $P_b(X_b, Y_b)$ is the tool nose curve right bound. A_{QiQi-1} is the area covered by edge Q_iQ_{i-1}. $A_2 = 0$	(B.11)
teF4	$A_1 = \frac{-r_\varepsilon}{4}(4[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)] + r_\varepsilon[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)] - 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1)) + \frac{1}{2}(Y_3 + Y_1)(X_3 - X_1)$ $A_2 = \frac{1}{2}(X_3Y_4 - X_4Y_3 + X_4Y_5 - X_5Y_4 + X_5Y_1 - X_1Y_5 + X_1Y_3 - X_3Y_1)$	(B.12)

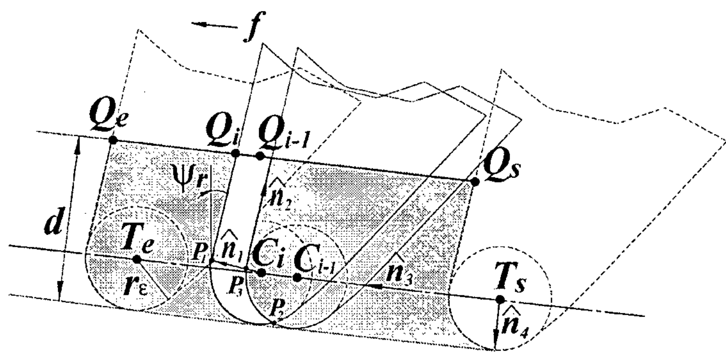
teF5	$A_1 = \frac{-r_\varepsilon}{4}(4[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)]$ $+ r_\varepsilon[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)]$ $- 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1)) + \frac{1}{2}(Y_3 + Y_1)(X_3 - X_1)$ $A_2 = \frac{1}{2}(X_1Y_5 + X_3Y_1 + X_4Y_4 + X_4Y_3 - X_5Y_1 - X_5Y_6$ $- X_1Y_3 - X_3Y_4) + A_{Q_iQ_{i-1}}$	(B.13)
teF6	$A_1 = \frac{-r_\varepsilon}{4}(4[Yc_i(\cos\theta_2 - \cos\theta_1) + Yc_{i-1}(\cos\psi_2 - \cos\psi_1)]$ $+ r_\varepsilon[\sin(2\theta_2) - \sin(2\theta_1) + \sin(2\psi_2) - \sin(2\psi_1)]$ $- 2r_\varepsilon(\theta_2 - \theta_1 + \psi_2 - \psi_1))$ $A_2 = 0$	(B.14)

Appendix C

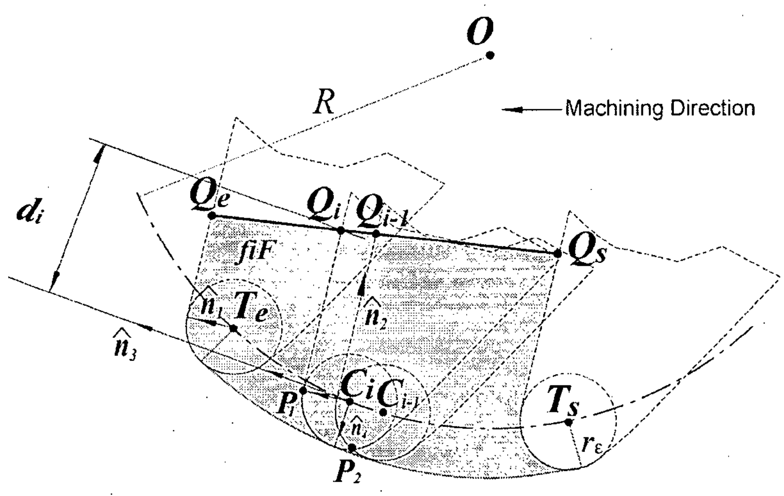
Engagement Boundary Identification in Geometric and Form Invariant Features

C.1 *teF* Boundary Identification

As described in Appendix B, for any *teF* shown in Figure B.1 appropriate boundary conditions have to be applied to the established equations to get the results. All the boundary conditions (point coordinates and angles) are derivable analytically and presented in this section.



(a) *teF* Extraction from *giF*



(b) *teF* Extraction from *fiF*

Figure C.1 *teF* Extraction within *gif* / *fiF*

Figure C.1 shows *teF* is extracted from geometric invariant feature (*giF*) and one example of form invariant feature (*fiF*). Since toolpath, workpiece boundary, and tool boundary edges can all be expressed as implicit or parametric equations, the intersections between tool successive positions, or in other words, boundary positions of a *teF*, can be derived analytically, and expressed as a function of C_i, i.e., tool nose center position along the toolpath.

For *teF2* and *teF3*, P_2, Q_i, Q_{i-1} need to be calculated, for *teF4* and *teF5*, P_1, P_2, P_3, Q_i, and Q_{i-1} need to be calculated. Besides, C_i, C_{i-1}, d, l_1, and l_2 need to be obtained for all the cases. In the rest of this section, deviation of these boundary points is presented.

For better understanding, the terminologies used are listed as follows:

T_s, T_e: the start and end positions of a toolpath;

O, R: the center position and radius of a circular toolpath;

S: the length of a toolpath (curve length for circular toolpath).

C_i, C_{i-1}: tool successive center positions on a toolpath;

Q_s, Q_e: the start and end positions of a workpiece boundary edge;

l: distance between C_i and C_{i-1};

Q, r_q: the center position and radius of a circular workpiece boundary edge;

Q_i, Q_{i-1}: workpiece boundary positions at one feed step;

C.1.1 Parametric Expression of Toolpath T_sT_e and Workpiece Boundary Q_sQ_e.

Toolpath can be a linear segment or an arc, expressed as follows,

$$C_i = \begin{cases} (1-t)T_s + tT_e & 0 \le t \le 1 \\ O + \begin{bmatrix} R\cos(t) & R\sin(t) \end{bmatrix} & \alpha_1 \le t \le \alpha_2 \end{cases} \tag{C.1}$$

α_1, α_2 are start and end angles of the toolpath, and are calculated from T_sO and T_eO.

The parameter t is
$$t = \begin{cases} i \cdot \Delta t = i \cdot {}^{f}\!/\!_{L} \\ \alpha_1 + i \cdot \Delta\alpha = \alpha_1 + i \cdot {}^{f}\!/\!_{R} \end{cases}, i = 1 \text{ to } n \tag{C.2}$$

Where n is the total feed steps in a toolpath, $n = {}^{L}\!/\!_{f}$, and i is the i[th] step.

Any workpiece boundary position Q_i can be expressed as follows;

$$Q_i = \begin{cases} (1-u)Q_s + uQ_e & 0 \le u \le 1 \\ Q + \begin{bmatrix} r_q \cos(u) & r_q \sin(u) \end{bmatrix} & \beta_1 \le u \le \beta_2 \end{cases} \tag{C.3}$$

It must be notes that the parameter u is calculated with respect to the toolpath parameter t, and it may not be uniformly incremented with feedrate.

C.1.2 Unit Vector Expression

As shown in Figure C.1, $\hat{n}_1$ is the unit vector of P_1C_i, $\hat{n}_3$ is the unit vector of toolpath (instantaneous feed direction in circular toolpath). $\hat{n}_5$ is the unit vector of C_iC_{i-1}. $\hat{n}_2$, $\hat{n}_4$, and $\hat{n}_6$ are the unit vectors perpendicular to $\hat{n}_1$, $\hat{n}_3$ and $\hat{n}_5$ respectively. They are expressed as follows.

$$\hat{n}_1 = [-\cos\psi_r, \sin\psi_r] \tag{C.4}$$

$$\hat{n}_2 = [\sin\psi_r, \cos\psi_r] \tag{C.5}$$

$$\hat{n}_3 = \begin{cases} \dfrac{T_e - T_s}{|T_e - T_s|} = \dfrac{T_e - T_s}{L} \\ \dfrac{\dot{C}_i(t)}{|\dot{C}_i(t)|} = \dfrac{[-R\sin(t), R\cos(t)]}{R} = [-\sin(t), \cos(t)] \end{cases} \tag{C.6}$$

$$\hat{n}_4 = \begin{cases} [-n_{3y}, n_{3x}] \\ [-\cos(t), -\sin(t)] \end{cases} \tag{C.7}$$

$$\hat{n}_5 = \frac{C_i - C_{i-1}}{|C_i C_{i-1}|} \begin{cases} \dfrac{T_e - T_s}{L} \\ [-\sin(t - \dfrac{f}{2R}), \cos(t - \dfrac{f}{2R})] \end{cases} \tag{C.8}$$

where $|C_i C_{i-1}| = l = 2R\sin(\frac{f}{2R})$

$$\hat{n}_6 = \begin{cases} \hat{n}_4 \\ [-\cos(t - \dfrac{f}{2R}), -\sin(t - \dfrac{f}{2R})] \end{cases} \tag{C.9}$$

C.1.3 P_1 Calculation

P_1 is the tangent point between tool side cutting edge and tool nose curve at tool current position (C_i) on the toolpath. This position is invariant with respect to C_i due to the rigid tool geometry. It can be expressed for all the cases:

$$P_1 = C_i + r_\varepsilon \cdot \hat{n}_1 \tag{C.10}$$

C.1.4 P_2 Calculation

P_2 is the intersection point between cutting edges of two tools at C_i and C_{i-1}. There are three cases: circle-circle intersection, circle-line intersection and line-circle intersection.

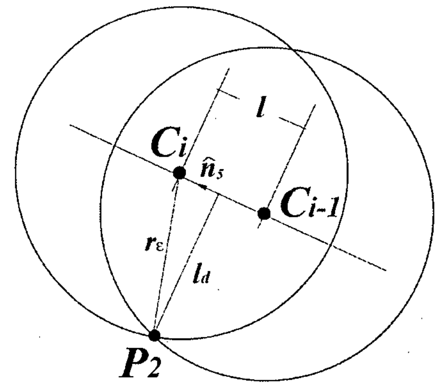


Figure C.2 Circle-Circle Intersection of P_2

For circle-circle intersection as shown in Figure C.2, P_2 can be expressed as:

$$P_2 = C_i - \frac{l}{2} \cdot \hat{n}_5 + l_d \cdot \hat{n}_6 \qquad \text{(C.11)}$$

where l_d is the distance from P_2 to C_iC_{i-1}, substitute l_d and L_{cc}, P_2 is:

$$P_2 = \begin{cases} C_i - \frac{f}{2} \cdot \hat{n}_3 + \sqrt{r_\varepsilon^{\,2} - \frac{f^2}{4}} \cdot \hat{n}_4 \\ C_i - R\sin(\frac{f}{2R}) \cdot \hat{n}_5 + \sqrt{r_\varepsilon^{\,2} - R^2 \sin^2(\frac{f}{2R})} \cdot \hat{n}_6 \end{cases} \qquad \text{(C.11)}$$

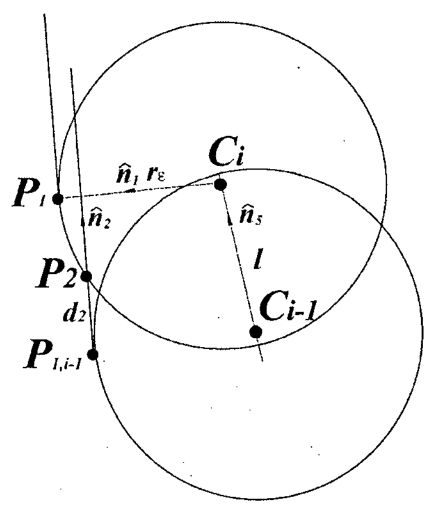


Figure C.3 Circle-Line Intersection of P_2

For circle-line intersection as shown in Figure C.3, P2 can be expressed as:

$$\begin{aligned} P_2 &= P_{1,i-1} + d_2 \cdot \hat{n}_2 \\ &= P_1 - l \cdot \hat{n}_5 + d_2 \cdot \hat{n}_2 \\ &= C_i + r_\varepsilon \cdot \hat{n}_1 - l \cdot \hat{n}_5 + d_2 \cdot \hat{n}_2 \end{aligned} \tag{C.12}$$

where d_2 is unknown. On the other hand, P_2 is on the circle centered at C_i,

$$\left|P_2 - C_i\right|^2 = r_\varepsilon^{\ 2} \tag{C.13}$$

Substitute equation (C.12) to equation (C.13), it becomes:

$$\left|r_\varepsilon \cdot \hat{n}_1 - l \cdot \hat{n}_5 + d_2 \cdot \hat{n}_2\right|^2 = r_\varepsilon^{\ 2} \tag{C.14}$$

Solving equation (C.14) to obtain two d_2, small one is required, it is:

$$d_2 = -\hat{n}_2 \cdot \vec{V} - \sqrt{(\hat{n}_2 \cdot \vec{V})^2 - (\left|\vec{V}\right|^2 - r_\varepsilon^{\ 2})} \tag{C.15}$$

Where, $\vec{V} = r_\varepsilon \cdot \hat{n}_1 - l \cdot \hat{n}_5$. Therefore, P_2 is expressed as:

$$P_2 = C_i + r_\varepsilon \cdot \hat{n}_1 - l \cdot \hat{n}_5 + d_2 \cdot \hat{n}_2 \tag{C.16}$$

Similarly, line-circle intersection of P_2 is calculated as follows and shown in Figure C.4. In which, P_b is the tangent point between tool nose curve and end cutting edge, $\hat{n}_b$ is the unit vector of P_bC_i, $\hat{n}_b'$ is the unit vector perpendicular to $\hat{n}_b$. They are expressed as follows.

$$P_b = C_i + r_\varepsilon \cdot \hat{n}_b \tag{C.17}$$

$$n_b = [\sin(\kappa_r), -\cos(\kappa_r)] \tag{C.18}$$

$$n_b' = [\cos(\kappa_r), \sin(\kappa_r)] \tag{C.19}$$

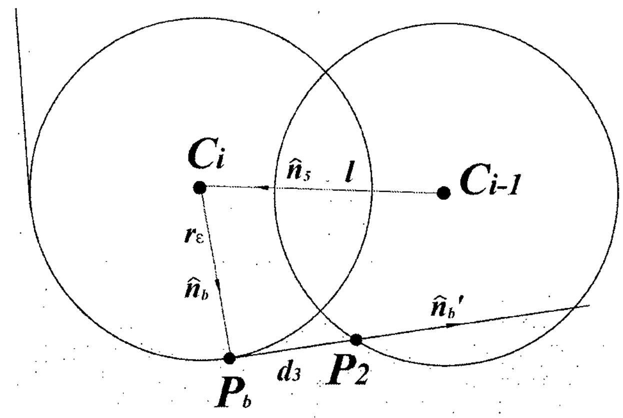


Figure C.4 Line-Circle Intersection of P_2

P_2 is formulated as:

$$P_2 = C_i + r_\varepsilon \cdot \hat{n}_b + l \cdot \hat{n}_5 + d_3 \cdot \hat{n}_b'$$

where

$$d_3 = -\hat{n}_b' \cdot \vec{W} - \sqrt{(\hat{n}_b' \cdot \vec{W})^2 - (|\vec{W}|^2 - r_\varepsilon^{\,2})} \qquad \text{(C.20)}$$

$$\vec{W} = r_\varepsilon \cdot \hat{n}_b + l \cdot \hat{n}_5$$

C.1.5 P_3 Calculation

P_3 is the intersection point of P_1C_i and the circle centered at C_{i-1}. It also can be circle-line intersection or line-line intersection as shown in Figure C.5 and Figure C.6. For the first case, it can be expressed in equation (C.21). After substituting P_1 and rearrange the equation, P_3 becomes equation (C.22). At the same time, P_3 is on the circle whose center is at C_{i-1} as shown in equation (C.23).

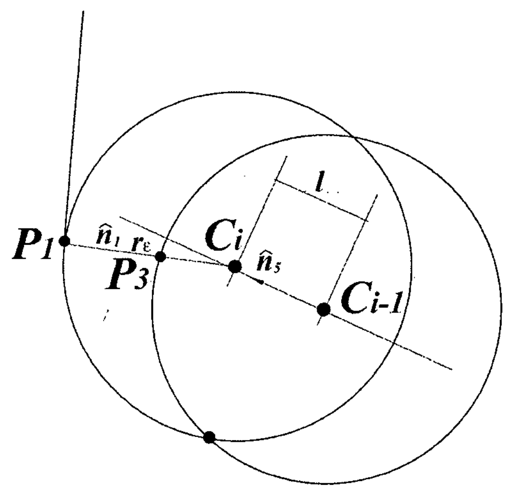


Figure C.5 Line-Circle Intersection of P_3

$$P_3 = (1-v)C_i + vP_1 \quad 0 \le v \le 1 \tag{C.21}$$

After substituting P_1:

$$P_3 = C_i + v \cdot r_\varepsilon \cdot \hat{n}_1 \tag{C.22}$$

$$\left|P_3 - C_{i-1}\right|^2 = r_\varepsilon^{\ 2} \tag{C.23}$$

Substituting (C.22) into (C.23), the equation becomes:

$$\begin{aligned} &\left|C_i - C_{i-1} + v \cdot r_\varepsilon \cdot \hat{n}_1\right|^2 = r_\varepsilon^{\ 2} \\ \Rightarrow\ &\left|l \cdot \hat{n}_5 + v \cdot r_\varepsilon \cdot \hat{n}_1\right|^2 = r_\varepsilon^{\ 2} \end{aligned} \tag{C.24}$$

Solving equation (C.24) gives two values of parameter v, since the left side intersection with circle C_{i-1} is required in this case, bigger v is taken to achieve this requirement as shown in equation (C.25).

$$\begin{aligned} &P_3 = C_i + v \cdot \hat{n}_1 \\ &v = \sqrt{(l \cdot \hat{n}_1 \cdot \hat{n}_5)^2 - L_{cc}^{\ 2} + r_\varepsilon^{\ 2}} - (l \cdot \hat{n}_1 \cdot \hat{n}_5) \end{aligned} \tag{C.25}$$

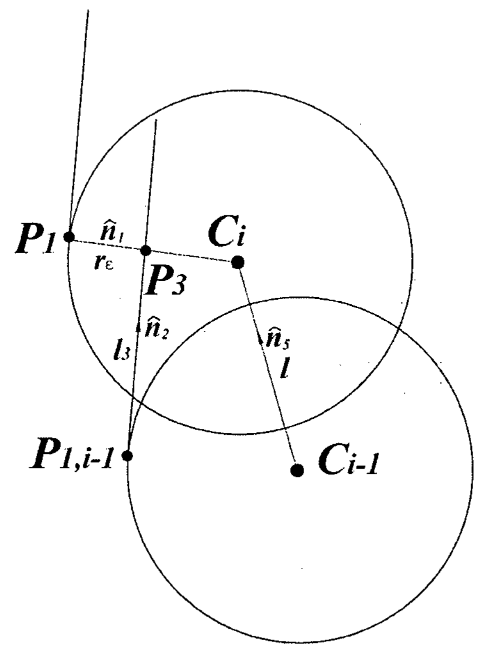


Figure C.6 Line-line Intersection of P_3

In this case, P_3 is the intersection point between line P_1C_i and $P_{1,i-1}P_2$.

$$\begin{cases} P_i = (1-t_1)C_i + t_1 P_1 \\ P_i = P_{1,i-1} + l_3 \hat{n}_2 \end{cases} \tag{C.26}$$

Substitute P_1 and $P_{1,i-1}$, l_3 and t_1 can be calculated.

$$C_i + t_1 \cdot r_\varepsilon \cdot \hat{n}_1 = C_i + r_\varepsilon \cdot \hat{n}_1 - l \cdot \hat{n}_5 + l_3 \cdot \hat{n}_2$$

$$l_3 = \frac{l(\hat{n}_{5x} \cdot \sin\psi_r + \hat{n}_{5y} \cdot \cos\psi_r)}{\cos^2\psi_r - \sin\psi_r} \tag{C.27}$$

Therefore, P3 is calculated as:

$$P_3 = C_i + r_\varepsilon \cdot \hat{n}_1 - l \cdot \hat{n}_5 + l_3 \cdot \hat{n}_2$$

$$l_3 = \frac{l(\hat{n}_{5x} \cdot \sin\psi_r + \hat{n}_{5y} \cdot \cos\psi_r)}{\cos^2\psi_r - \sin\psi_r} \tag{C.28}$$

C.1.6 Workpiece boundary Q_i Calculation

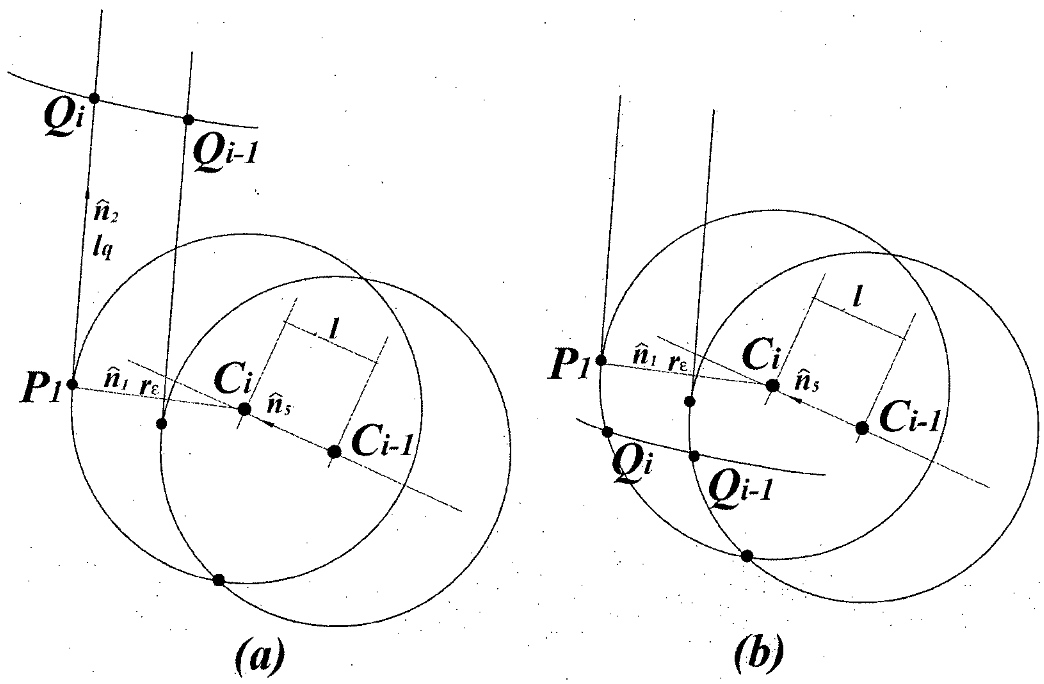


Figure C.7 Workpiece Boundary Point of Q_i

From Figure C.6, workpiece boundary position Q_i may intersects with the straight cutting edge of C_i, or tool nose curve of C_i. The relations with C_i in these two cases as shown in equation (C.26):

If Q_i intersects with the side cutting edge of the tool

$$\begin{aligned} Q_i &= P_1 + lq \cdot \hat{n}_2 \\ &= C_i + r_\varepsilon \cdot \hat{n}_1 + lq \cdot \hat{n}_2 \end{aligned} \tag{C.26}$$

If Q_i intersects with the tool nose curve:

$$|Q_i - C_i|^2 = r_\varepsilon{}^2$$

Together with equation (C.3), two equations have two unkowns, l_q and u can be solved. Consequently, Q_i can be solved, and also expressed as a function of C_i.

C.1.7 Depth of Cut *d* Calculation

d is defined as the distance from Q_i to instantaneous feed direction (n_3):

$$d = |Q_i - C_i| \times \hat{n}_3 + r_\varepsilon \tag{C.27}$$

Similarly, the distance between P_1, P_2 and n_3 are:

$$l_1 = |P_1 - C_i| \times \hat{n}_3 + r_\varepsilon \tag{C.28}$$

$$l_2 = |P_2 - C_i| \times \hat{n}_3 + r_\varepsilon \tag{C.29}$$

C.2 Recursive expression of *teF* boundaries

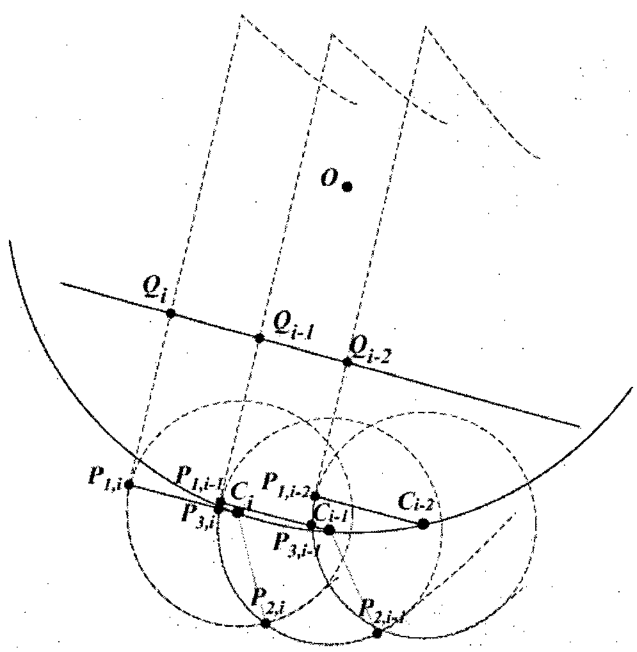


Figure C.8 Recursively Expression of Boundaries

A successive express of C_i can be obtained from equation (C.1) as shown in equation (C.30), and a simplified expression is in (C.31).

$$C_i = \begin{cases} C_{i-1} + \Delta t(T_e - T_s), \textit{ line toolpath} \\ O + \left[V_x \cos(\Delta t) - V_y \sin(\Delta t) \quad V_y \cos(\Delta t) - V_x \sin(\Delta t)\right], \quad \textit{arc toolpath} \end{cases} \tag{C.30}$$

$$=> \qquad C_i = \begin{cases} C_{i-1} + f \cdot \hat{n}_3, \textit{ line toolpath} \\ O + \vec{N}_i, \quad \textit{arc toolpath} \end{cases} \tag{C.31}$$

where, $V = [V_x \; V_y] = C_{i-1} - O$, tool increment $\Delta t = \begin{cases} f/L, & \textit{line} \\ f/R, & \textit{arc} \end{cases}$.

From equation (C.31), all boundary positions can be expressed as the recursive equations with respect to their previous positions.

$P_{1,i}$ and $P_{1,i-1}$	$P_{1,i} = C_i + r_\varepsilon \cdot \hat{n}_1$ $= \begin{cases} C_{i-1} + r_\varepsilon \cdot \hat{n}_1 + f \cdot \hat{n}_3, \textit{ line toolpath} \\ O + r_\varepsilon \cdot \hat{n}_1 + \vec{N}_i, \quad \textit{arc toolpath} \end{cases}$ Therefore, $P_{1,i} = P_{1,i-1} + f \cdot \hat{n}_3$ line toolpath $\begin{cases} Xp_{1,i} = E_1 + Xp_{1,i-1} \cdot \cos(f/R) - Yp_{1,i-1} \cdot \sin(f/R) \\ Yp_{1,i} = E_2 - Xp_{1,i-1} \cdot \sin(f/R) + Yp_{1,i-1} \cdot \cos(f/R) \end{cases}$, *circular toolpath* where, $E_1 = (1 - \cos(f/R))(O_x + r_\varepsilon \cdot \hat{n}_{1x}) + \sin(f/R)(O_y + r_\varepsilon \cdot \hat{n}_{1y})$ $E_2 = (1 - \cos(f/R))(O_y + r_\varepsilon \cdot \hat{n}_{1y}) + \sin(f/R)(O_x + r_\varepsilon \cdot \hat{n}_{1x})$	(C.32)

$P_{2,i}$ and $P_{2,i-1}$	For linear toolpath, $P_{2,i} = P_{2,i-1} + f \cdot \hat{n}_3$ For circular toolpath, $P_{2,i} = function(P_{2,i-1})$ Due to the complicity of the equation and the limitation of the space, the detailed expression will not be given in this table.	(C.33)
$P_{3,i}$ and $P_{3,i-1}$	For linear toolpath, $$P_{3,i} = P_{3,i-1} + f \cdot \hat{n}_3$$ For circular toolpath $$\begin{cases} Xp_{3,i} = F_1 + Xp_{3,i-1} \cdot \cos(f/R) - Yp_{3,i-1} \cdot \sin(f/R) \\ Yp_{3,i} = F_2 - Xp_{3,i-1} \cdot \sin(f/R) + Yp_{3,i-1} \cdot \cos(f/R) \end{cases}, \textit{circular toolpath}$$ where, $$F_1 = (1-\cos(f/R))(O_x + v \cdot \hat{n}_{1x}) + \sin(f/R)(O_y + v \cdot \hat{n}_{1y})$$ $$F_2 = (1-\cos(f/R))(O_y + v \cdot \hat{n}_{1y}) + \sin(f/R)(O_x + v \cdot \hat{n}_{1x})$$	(C.34)

Using the successive expression, especially for linear toolpath, boundary positions P_i can be calculated using P_{i-1}. The computational speed is significantly increased.