INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA

UMI®
800-521-0600

MODEL BASED DIE CAVITY MACHINING SIMULATION METHODOLOGY

By

BEHNAM MOETAKEF-IMANI, B. Sc., M. Sc. (Mechanical Engineering)

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Doctorate of Philosophy

McMaster University

Model Based Die Cavity Machining Simulation Methodology

DOCTORATE OF PHILOSOPHY (1997) McMaster University

(Mechanical Engineering) Hamilton, Ontario

TITLE: Model Based Die Cavity Machining Simulation Methodology

AUTHOR: Behnam Moetakef-Imani

B. Sc. (Mechanical Engineering) Sharif University of Technology, Tehran, Iran

M. Sc. (Mechanical Engineering) Technical Faculty, Tehran University, Tehran, Iran

SUPERVISOR: Dr. M.A. Elbestawi, Professor

Department of Mechanical Engineering

McMaster University

NUMBER OF PAGES: xiv, 126

Abstract

The current emphasis of CAD/CAM technology, in particular the die and mold manufacturing sector, is to verify and optimize the NC code in terms of the productivity and machining accuracy prior to the actual machining. Also, there is a trend towards enhancing the on-line control strategies with the technological data. However, the intricate geometry with sculptured surfaces which is found nowadays in most of dies and molds, together with relative high material hardness, makes the NC verification/optimization process a very demanding and difficult task. In this regard, the development of model-based milling simulation methodology with functions of geometrical and physical modeling is a fundamental advancement in improving productivity, accuracy, and automation.

In the course of the thesis presented, a new model-based simulation methodology for die cavity milling operations is developed. The comprehensive and realistic simulation of the milling process is implemented in the form of an integrated geometrical/physical/interaction environment. The geometric sub-environment constructs the volume swept by the cutter (or the surface swept by the cutting edge), accurately updates the model of the stock as the cutter removes the material, and automatically extracts the immersion geometry from the updated model. Based on the knowledge of

the material remaining on the surface, two- and three-axis tool paths are computed. In the semi-finishing operation, a three-axis milling strategy is developed to control the scallop height remaining on the surface and avoid cutting with the dead zone of the cutter. The physical/mechanical sub-environment, integrated with the geometric counterpart, consists of a newly developed force model as well as a mathematical model of tool-structure dynamics. The integrated environment can accurately simulate the closed loop of the machine tool-cutting process. It predicts the instantaneous cutting forces, tool tip deflections, and the onset of chatter vibrations. The finishing operation is simulated based on the true path of the cutting edge. The method developed is able to construct the geometric model of feed marks and scallops remaining on the surface.

Experimental measurements confirm the validity of the implemented methodology. First, the chip geometry is extracted, the part is updated, and the NC codes are verified for two- and three-axis milling of free-form surfaces. Second, the instantaneous cutting forces, tool tip deflections, the onset of chatter vibrations are predicted and compared with the experimental measurements. Next, a new three-axis milling strategy is implemented to control the scallop height and avoid cutting with the dead zone of the cutter. Lastly, the model of feed marks and scallop height are constructed, the feed mark profile is extracted and compared with the measurement.

The developed system enhances the capabilities of the CAM/CAM systems in terms of increased productivity, improved machining accuracy, and heightened the level of automation.

Contents

Abstract iii

Acknowledgments xiv

1 Introduction 1

1.1 Overview 1

1.2 Current die and mold manufacturing practices 2

1.3 Limitations of available CAD/CAM systems 4

1.4 Thesis focus 5

2 Literature review 7

2.1 Geometric modeling computational techniques 7

2.1.1 Wire-frame modeling 7

2.1.2 Surface modeling 8

2.1.3 Solid modeling 9

2.1.4 Free-form surface representation 17

2.2 Geometric modeling of milling operations 19

2.2.1 Milling geometric simulation requirements 20

2.3 Physical simulation of milling operations 22
2.3.1 Milling force model 22
2.3.2 Machine tool dynamics 24
2.3.3 Closed loop of machine tool-cutting process 24
2.4 Simulation systems 25

3 Model-based milling simulation methodology **27**
3.1 Basic concept 27
3.2 Geometric modeling 28
3.3 Physical modeling 30
3.4 Interaction modeling 31

4 Two-axis roughing operation **32**
4.1 Optimum tool path pattern 33
4.2 Geometric simulation of flat-end milling 37
4.2.1 Two-axis swept volume generation 38
4.2.2 Part updating in roughing operation 41
4.2.3 Instantaneous undeformed chip geometry 44
4.3 Physical simulation of flat-end milling 47
4.3.1 Force model for flat-end milling 47
4.3.2 Tool and structure dynamics 49
4.3.3 Closed loop of machine tool-cutting process 53
4.3.4 Identifying onset of chatter vibrations 54
4.4 Experimentation and verification 55
4.4.1 Experimental setup 56

4.4.2 Mechanistic force model calibration 57
4.4.3 Onset of chatter vibrations 57
4.5 Feed scheduling 58
4.6 Summary 60

5 Two- and three-axis semi-finishing **62**
5.1 Tool path based on updated B-rep model 63
5.1.1 Two-axis tool path 63
5.1.2 Scallop height control 65
5.2 Geometric simulation of ball-end milling 69
5.2.1 Three-axis swept volume generation 69
5.2.2 Tool path verification 75
5.2.3 Instantaneous undeformed chip geometry 78
5.3 Physical simulation of ball-end milling 82
5.3.1 Mechanistic force model development 84
5.3.2 Model calibration 87
5.4 Experimentation and verification 91
5.5 Feed scheduling 95
5.6 Summary 96

6 Finishing operation **97**
6.1 B-rep model of feed marks and scallops 98
6.2 Feed mark profile 100
6.3 Summary 101

7 Summary 104

A Tolerances in the B-rep geometric modeler 109

B Modal testing 113

References 117

List of Figures

1.1 Comparison of production costs between large and low scale productions 3

2.1 Surface modeling data structure[16] 8

2.2 Octree representation [35] 11

2.3 CSG data structure [35] 12

2.4 B-rep data structure [35] 13

2.5 Manifold and non-manifold solids [30] 14

2.6 B-rep winged-edge data structure [30] 15

2.7 EDGEs, COEDGEs, and LOOPs in a B-rep model 15

2.8 Implementation of solid BODY [51] 16

3.1 Model-based simulation environment 28

3.2 Closed loop of the machining unit 30

3.3 B-rep model of chip in ball-end milling 31

4.1 Pocket machining and bisector diagram[45] 34

4.2 Bisector diagram enhanced with orientations [28] 34

4.3 Efficient tool path patterns [17] 35

4.4 Proposed tool path for pocketing operation 36

4.5 Feed marks or cusps left on machined surface 38
4.6 Flat-end mill geometry 38
4.7 two-axis swept volume: rigid tool 39
4.8 Deflection of tool profile 41
4.9 Instances of tool profile along two-axis tool path 41
4.10 Two-axis swept volume: flexible tool 42
4.11 One quarter of die with sculptured surfaces 43
4.12 Updated part for the first one-way pass 43
4.13 B-rep model of rough-cut part 44
4.14 Oriented semi-circles along the tool path 46
4.15 Incut segments of semi-circles along the tool path 46
4.16 Variation of immersion geometry along the tool path 47
4.17 Differential cutting forces and undeformed chip thickness 49
4.18 Spindle and tool mode shapes [61] 50
4.19 Lumped mass-damper-spring model 50
4.20 Standardized probability density function [11] 55
4.21 Experimental setup 56
4.22 Flat-end milling model calibration 58
4.23 Onset of Chatter vibrations 59

5.1 Two-axis tool path and left scallops 63
5.2 Scallop height distribution for two-axis tool path 64
5.3 Dead zone constraint 65
5.4 Scallop height control 66

5.5 Scallop height distribution for three-axis tool path 68
5.6 Scallop height distribution after performing extra tool path 68
5.7 Three-axis swept volume and its boundary faces 71
5.8 Global and local coordinate system 73
5.9 Semi-finished step-by-step updated part 76
5.10 Semi-finished updated part with unified FACEs 77
5.11 Ball-end mill with constant lead design 79
5.12 Instantaneous in-cut segment extraction 80
5.13 Variation of chip thickness for three-axis feed 81
5.14 Chip thickness: comparison of simplified and accurate representation 83
5.15 Unit vectors on rake face and cutting edge 84
5.16 Differential chip contact area 85
5.17 Half-immersion integration limits 88
5.18 Model calibration 89
5.19 Variations of cutting coefficients 90
5.20 Variation of chip flow angle 90
5.21 Three-axis immersion geometry and in-cut segments 91
5.22 Integration limits of three-axis ball-end milling 92
5.23 Predicted and measured cutting forces for 10 and 15 degrees uphill angles 93
5.24 Predicted and measured cutting forces for 20 and 25 degrees uphill angles 94

6.1 Insert geometry 98
6.2 Surface swept by the cutting edge with coupled cutting and three-axis feed motions 99

6.3 B-rep model of feed marks and cross feed scallops 100

6.4 Surface roughness profile 102

6.5 Variation of peak-to-valley R_{max} with feed and tool diameter 102

A.1 Modeling world precision and range 110

A.2 Improving fitting tolerance in interpolating a semi-circle 112

B.1 General form of measured transfer function: real part 115

List of Tables

1.1 Cost structure in die/mold manufacturing 4

4.1 Flat-end milling cutting coefficients 57

5.1 Ball-end mill geometry 79

B.1 Modal parameters of tool-spindle assembly 116

Acknowledgments

The author wishes to express his sincere gratitude to his research supervisor, Professor M.A. Elbestawi, for his guidance, encouragement, and support throughout the course of this research.

Gratitude is also expressed to the members of the supervisory committee : Dr A.D. Spence, Dr. J. Vlachopoulos, and Dr. M. Sklad for their continuing interest. The author also wishes to express his gratitude to his colleagues at IMMRC and the staff at the Department of Mechanical Engineering, McMaster University for their help at various occasions.

The financial supports jointly provided by Iran's Ministry of Culture and Higher Education for the period of four years and the Department of Mechanical Engineering through a Teaching Assistantship and a Departmental Scholarship are gratefully acknowledged.

Finally, I would like to thank my family, my wife and my daughter, without their support and undertanding none of this would have been possible.

Chapter 1

Introduction

1.1 Overview

The milling operation is a manufacturing process with a long history and prominent importance. With the development of Numerical Control (NC) milling machines, which make the machining of the intricate geometry with sculptured surfaces possible, milling operations are widely used for the production of complicated, high-precision, and low-quantity parts such as dies and molds.

In three-axis NC milling, a rotating cutting tool follows a sequence of movements and instructions specified by the NC code and cuts the sculptured shape out of a block of metal (the stock). The NC code includes cutter location (CL) data, which is a long list of coordinates. The cutter moves to the first coordinate and starts to cut the material along the path to the second and third locations and so on, until it reaches to the last location.

The NC codes are generated by NC programmers using various commercial CAD/CAM

systems. The programmers, provided with a geometric representation of the part, generate a sequence of movements which are supposed to machine the part from the stock. They also consult machining data handbooks to choose safe and conservative cutting conditions such as feeds and cutting speeds.

NC codes, CL data and cutting conditions, are prone to errors. For example, too deep cuts or heavy chip loads can be encountered which result in surface gouging or excessive cutting forces, respectively. Thus, an important practical problem in this regard is: "*Given a geometric representation of a surface (the design surface) and the corresponding NC code, can the NC code generate the design surface within prescribed tolerances*?" Once the above problem is solved successfully, the focus shifts towards the issues of *efficiency* and *decreasing manufacturing costs.*

The early approach to verification and optimization of NC codes was to perform proof-running tests on soft materials. This is still a popular technique. However, only large geometrical errors can be found in this way. On the other hand, the intricate geometry and relatively high material hardness add to the complexity of the die and mold NC verification/optimization process.

1.2 Current die and mold manufacturing practices

The practice of machining a die cavity bounded by sculptured surfaces comprises the following sequence of metal removal operations[76][37]:

1) Drilling an initial hole for subsequent milling operations
2) Two-axis rough cutting with large flat-end mill (pocketing operation)
3) Semi-finish cutting with large ball-end mill (shoulder removing operations)

4) Finish cutting with small ball-end mill

5) Electrical discharge machining (EDM) for deep and/or sharp internal corners

6) Polishing

Die and mold manufacturing is a low scale production industry. Figure 1.1[48] compares die and mold manufacturing with large scale production industries. Clearly, the die and mold makers have less opportunity to economize on the material costs than on the controllable manufacturing costs.

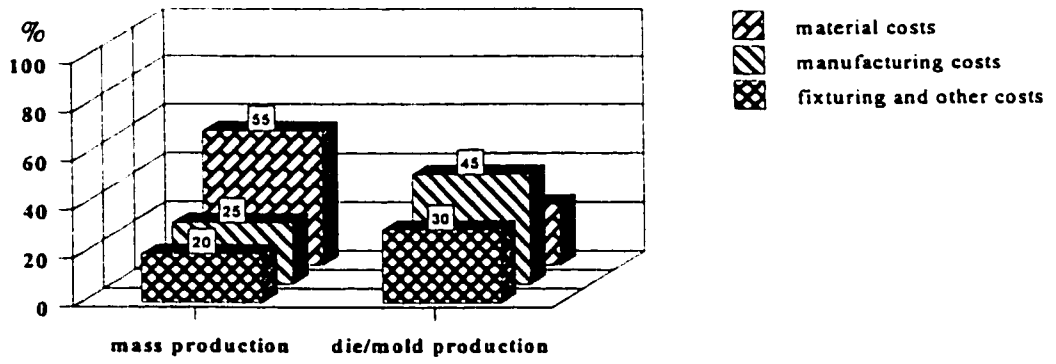


Figure 1.1: Comparison of production costs between large and low scale productions

Typically, the cost structure of the manufacturing operations involved in the conventional milling of a medium sized die or mold would break down as shown in Table 1.1[48]. The level of cutting forces in roughing and semi-finishing operations, which represent about 40 percent of the total manufacturing costs, is very high. Therefore, the NC programmers are forced to select very conservative cutting conditions based on the maximum chip load. This approach is clearly inefficient. In the finishing, the main concerns are the machining errors, tool life, and the surface roughness on the heat-treated steel material. The main trend in finishing is towards five-axis high

Operation	Time (hours)	Hourly rate ($/h)	Total cost ($)	Total cost (%)
Roughing	30	95	2850	12
Semi-finishing	60	95	5700	25
Finishing	60	95	5700	25
Hand polishing	75	50	3750	16
Rework/fitting	100	50	5000	22
Total			23000	100

Table 1.1: Cost structure in die/mold manufacturing

speed milling operations. In practice, the important limiting factor is the availability of a comprehensive CAD/CAM system with functions for extracting the chip load, predicting physical/mechanical machining parameters, and optimizing the NC codes.

1.3 Limitations of available CAD/CAM systems

In the die and mold manufacturing industry, various commercial CAD/CAM systems are widely used to generate NC codes. Although the current state of CAM technology is capable of generating tool paths for free-form (or sculptured) surfaces with prescribed tolerances, their role in achieving reliable and efficient NC codes is still limited.

First, the NC code generation/verification performed by available systems is limited to checking nominal geometric dimensions. However, machining operations of free-form surfaces require comprehensive step-by-step geometric verifications of free-form geometries. Also, variations from nominal geometry as a result of the tool tip deflection, the cutting edge wear, and positioning uncertainties must be taken into

account.

Second, these systems provide little assistance to the part programmer in terms of choosing the proper cutting conditions such as feed, immersion geometry, and spindle speed. In these systems, the NC program for milling a given surface is normally generated with a constant spindle speed and feed. Therefore, the NC programmer is forced to use conservative spindle speeds and feeds based on the worst immersion geometry.

Finally, the available systems do not take the physical/mechanical aspects of the milling process into consideration. However, the physical/mechanical parameters such as cutting forces are central for generating reliable and efficient NC codes. The comprehensive and realistic prediction of the physical/mechanical parameters requires a model-based simulation environment which can model the milling process at different levels of accuracy.

1.4 Thesis focus

This thesis proposes a new model-based simulation methodology in order to accurately simulate the two- and three-axis milling operations of dies and molds with free-form surfaces. The implemented methodology is twofold. The first part is the geometric modeling of the milling operations with varying levels of accuracy and complexity. This part can serve two purposes: 1) verifying the tool paths and updating the part as the cutter removes material, and 2) computing the chip geometry. The physical/mechanical simulation, the second part, takes into account the mechanics of the cutting process as well as the dynamics of the tool and structure. Using the

extracted chip geometry, this part computes the physical/mechanical parameters required. Depending on the application, the physical/mechanical module simulates the machining process with different levels of accuracy.

The implemented methodology is used to optimize the NC code generated for flat-end milling and ball-end milling operations. In this regard, the main objective is to maximize metal removal rate(MRR) which is subjected to the machining error constraint and the chatter vibration constraint. The geometric module accurately simulates two- and three-axis milling operations, updates the part, and extracts the chip geometry. The method accurately predicts the instantaneous cutting forces as well as maximum tool tip deflections. Also, it anticipates the onset of chatter vibrations. At the highest level of geometric simulations, feed marks and scallops on the finished part surface are constructed.

Chapter 2

Literature review

2.1 Geometric modeling computational techniques

In general, three-dimensional (3D) geometric modeling techniques can be categorized into three groups [35][79]: 1) Wire-frame modeling which represents 3D objects by its edges. 2) Surface modeling which defines mathematical surfaces bounded by the edges. 3) Solid modeling which defines a valid and unambiguous 3D object. In the following sections a brief description of each group is presented first, with an emphasis on the solid modeling techniques. Then, the most accepted representation scheme for the free-form surfaces is reviewed.

2.1.1 Wire-frame modeling

A wire-frame model consists of points, lines, conics, and free-form curves. This model is the simplest geometric model that can represent an object in the computer. Also, the wire-frame can be used as a basis to constructed more complicated models such

as surface or solid models. However, the wire-frame of complex objects is incomplete and ambiguous since there is no differentiation between the inside and outside of the part representation. Thus, it cannot be used for CAM applications and analyses.

2.1.2 Surface modeling

A surface model is a geometric representation of an open object. It is a more complete and less ambiguous representation than the wire-frame model as it provides the information on surface(s) connecting the object boundaries. Thus, the surface modeling scheme is used to generate NC tool paths. Figure 2.1 depicts the needed relationship among the geometric entities of a surface modeler.

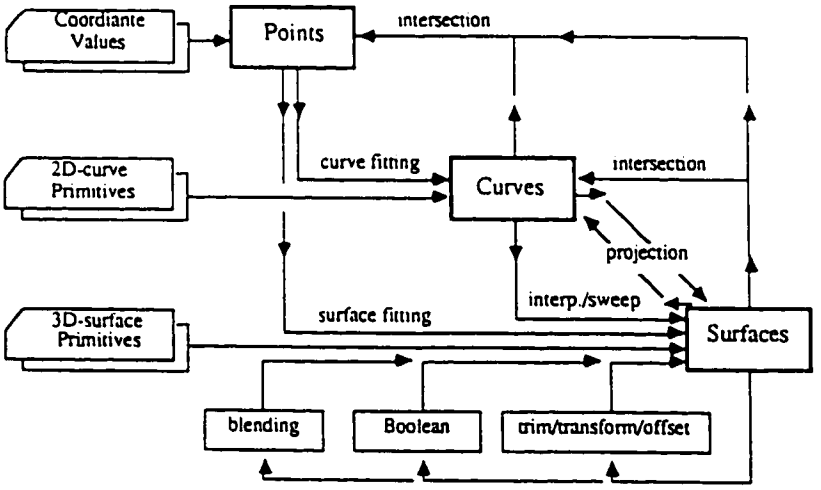


Figure 2.1: Surface modeling data structure[16]

The free-form or sculptured surface representation in car body design, ship hull design, and aircraft industry led to the field of Computer Aided Geometric Design(CAGD) [22]. This field covers many useful classes of parametric surfaces and

algorithms for design, analysis, and manipulation. However, surface modeling lacks topology (the relationship between different geometric entities), a deficiency which greatly complicates modifications of surface models.

2.1.3 Solid modeling

The most important aspect of the solid modeling is that the software can tell if a given object is forming a closed volume. In other words, it is possible to automatically check that if the representation of the object is information-ally complete, valid, and unambiguous. A complete geometric representation of an object is one that enables points in space to be classified relative to the object. This classification is called spatial addressability.

Thus, the part generated by solid modeling can be used for CAM applications such as machining simulations. For example, in the milling simulation the part is updated for each stage of machining and its validity can be automatically checked. Also, it is possible to automatically locate the interference using the solid modeling. This aspect can help the tool path generation softwares to locate the surface gouging portion of the tool path and take corrective actions.

As the solid models are less abstract and more realistic, it is possible to investigate their behavior under a variety of physical conditions. In fact, for simulations digital solid models are superior to the physical prototype. The solid modeling makes the massive interactive modifications possible. As the hardware comes down in price, the solid modeling simulations are less expensive than the actual simulations on prototypes.

In solid modeling many different representation schemes have evolved. Among

them, the following approaches are most widely accepted[35][79][30]: 1) Spatial enumeration by means of octrees; 2) Constructive solid geometry (CSG); and 3) Boundary representation (B-rep). Among these techniques, B-rep is used almost exclusively in CAD/CAM applications where precise geometries are required.

The spatial enumeration approach divides the space by means of a grid of cubes. The object is represented by the cubes that completely stay within the model. To precisely approximate the boundary of the model, smaller and smaller cubes (or grid size) are required. The data structure used by the spatial enumeration is called octree, which is derived from the fact that in a regular division of space, each cube can be divided into eight cubes. Thus, in an octree at each node eight-way branching tree structure is defined (Figure 2.2). The root node defines the entire universe and at the next level, eight children represent the eight cubical octants of the universe. The subdivision can continue to the needed level of resolution. Octree nodes can contain (Figure 2.2): 1) No part of the model which is indicated by white color; 2) The whole model which is indicated by black color; or 3) Some part of the model which is indicated by gray color.

The important aspect of the octree data structure is that it sorts the space; thus, the model can be searched efficiently. The recursive nature of the octree results in algorithms that are simple, compact, and fast. Also, the almost exclusive use of the integer arithmetic (rather than the floating-point arithmetic) facilitates low cost implementations. However, the octree is an approximate representation and unevaluated. If a higher level of precision is required, a large amount of the system space must be assigned. This representation is suitable for on-the-fly computations, such as mass properties where computed as needed and discarded.

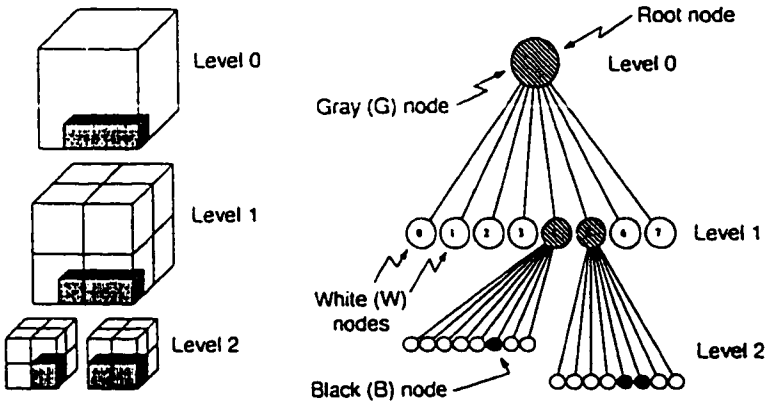


Figure 2.2: Octree representation [35]

In the CSG representation scheme, an object is constructed using Boolean operations (unite, subtract, and intersect) which combine simple solid objects (primitives) such as blocks, cylinders, spheres. CSG objects are usually stored in a tree data structure which is called CSG tree (Figure 2.3). In the CSG tree, leaf and branch nodes represent the primitives and boolean operations, respectively. In addition, the CSG tree can contain transformation nodes to change the location and orientation of an object. Non-geometric information such as material type can also be attached to the nodes.

The CSG representations are very compact and always valid. In addition, the implementation of Boolean operations is rather simple. The unevaluated nature of CSG can also be beneficial. For instance, the decisions of how the intersection curves are to be represented with particular tolerances can be postponed until they are actually needed. However, the CSG representations are limited to the geometry

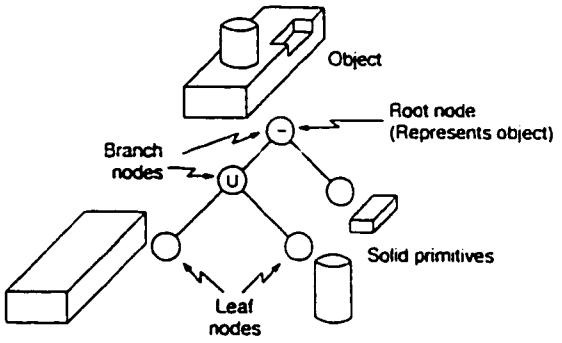


Figure 2.3: CSG data structure [35]

defined by the primitives. For example, in some CSG modelers the primitives are limited to shapes with quadratic surfaces and the modeler cannot support free-form surfaces. Also, the unevaluated nature of CSG limits its applications. For example, surfaces, edges, and vertices are not evaluated and are unavailable in closed forms. This limits the CAM applications where these entities are needed.

The boundary representations (B-rep) scheme is based on the notion that a 3D object is enclosed by a set of faces. The faces are parts of closed and orientable surfaces [30][79]. Where the surfaces intersect, curves are formed and similarly, where the curves intersect, the points are formed. Thus, the basic geometric entities needed are points, curves, and surfaces. The surfaces (and curves) such as plane, cylinder, and so forth (or line, circle, and so forth) are unbounded geometries. The solution adopted by the B-reps is to explicitly define the boundaries of surfaces (and curves). The solution requires the relationships between different geometric elements. For instance, which curves bound a surface (or which points bound a curve). This sort of information is known as topology which can be considered a graph between nodes

representing individual points, bounded curves, and bounded surfaces. In this regard, the following topological entities (represented by uppercase names) are defined: 1) VERTEX which represents a point at the boundary of a curve; 2) EDGE which represents a segment of curve; and 3) FACE which represents a portion of a surface. For example, in the B-rep graph illustrated in Figure 2.4, the nodes are FACEs and the connections represent the common EDGEs.

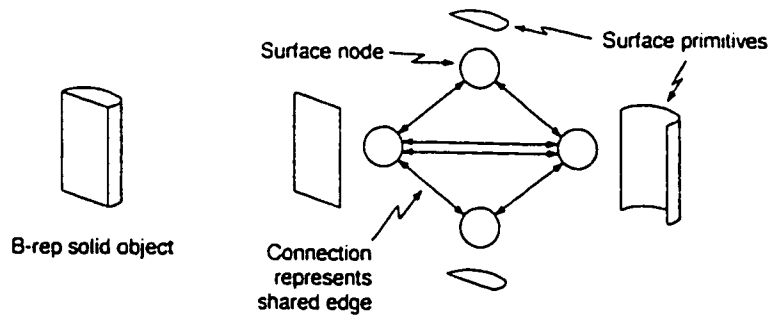
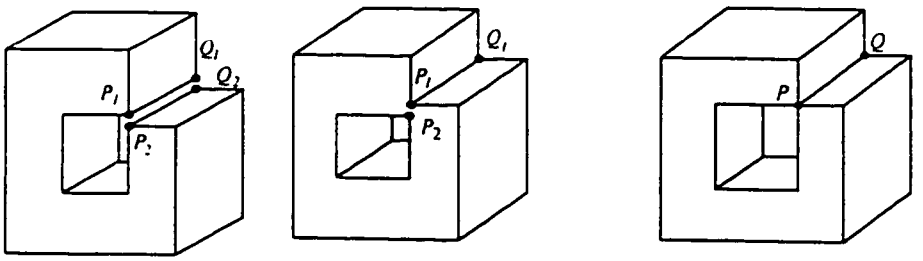


Figure 2.4: B-rep data structure [35]

In theory, solid B-reps can be separated into manifold and non-manifold. A manifold B-rep consists of manifold surfaces. A manifold surface can be deformed locally into a plane without tearing. In other words, it is possible to identify separate points with each other after the deformation [30]. All the physically realizable solid objects are manifold objects; however, this is not true when modeling solid with B-rep. For instance, a Boolean operation on manifold solid can result in a non-manifold solid (Figure 2.5). In the implementation, the manifold B-rep must obey the following rules : 1) Every EDGE must lie between two FACEs ; 2) FACEs and EDGEs do not self-intersect; and 3) Every entity in the model is bounded.



a- manifold solids b- non-manifold solid

Figure 2.5: Manifold and non-manifold solids [30]

In early B-rep implementations, the winged-edge data structure was introduced (Figure 2.6). It described manifold polyhedral objects by three tables for VERTICEs, EDGEs, and FACEs. Its important aspect is that the EDGE record contained the directional information regarding neighboring edges which lay on the two FACEs adjacent to it (Figure 2.6).

The geometric kernel [52], used in the current research, associates each EDGE with two COEDGES for manifold objects in order to separate the orientation role of the EDGE. Thus, the topological role of recording the adjacency is done by the EDGE and the orientation role is assigned to the COEDGE. The COEDGEs are grouped together to form a LOOP which represents the FACE boundary (Figure 2.7).

FACEs are also grouped together to form a SHELL. In some cases, the B-rep model can have two or more SHELLs; for example, when there is a void in the solid.

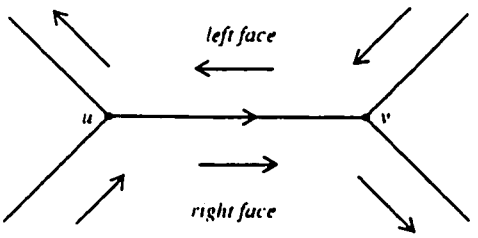


Figure 2.6: B-rep winged-edge data structure [30]

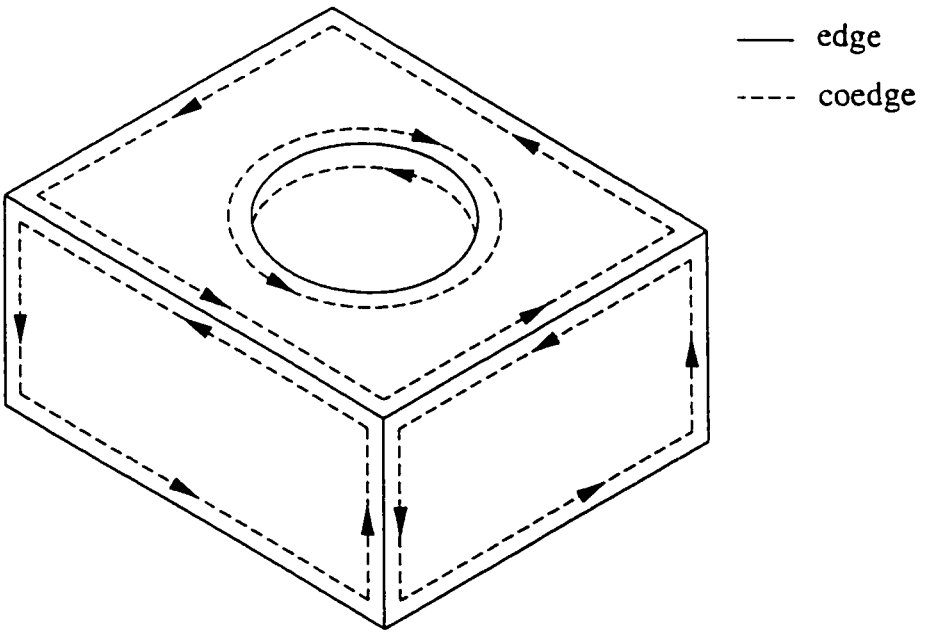


Figure 2.7: EDGEs, COEDGEs, and LOOPs in a B-rep model

Similarly, the SHELLs can be grouped together to form separable pieces of an object which is called a LUMP. Lastly, LUMPs are grouped together to form a topological entity called a BODY. Figure 2.8 illustrates the implementation of the solid BODY in the geometric kernel used.

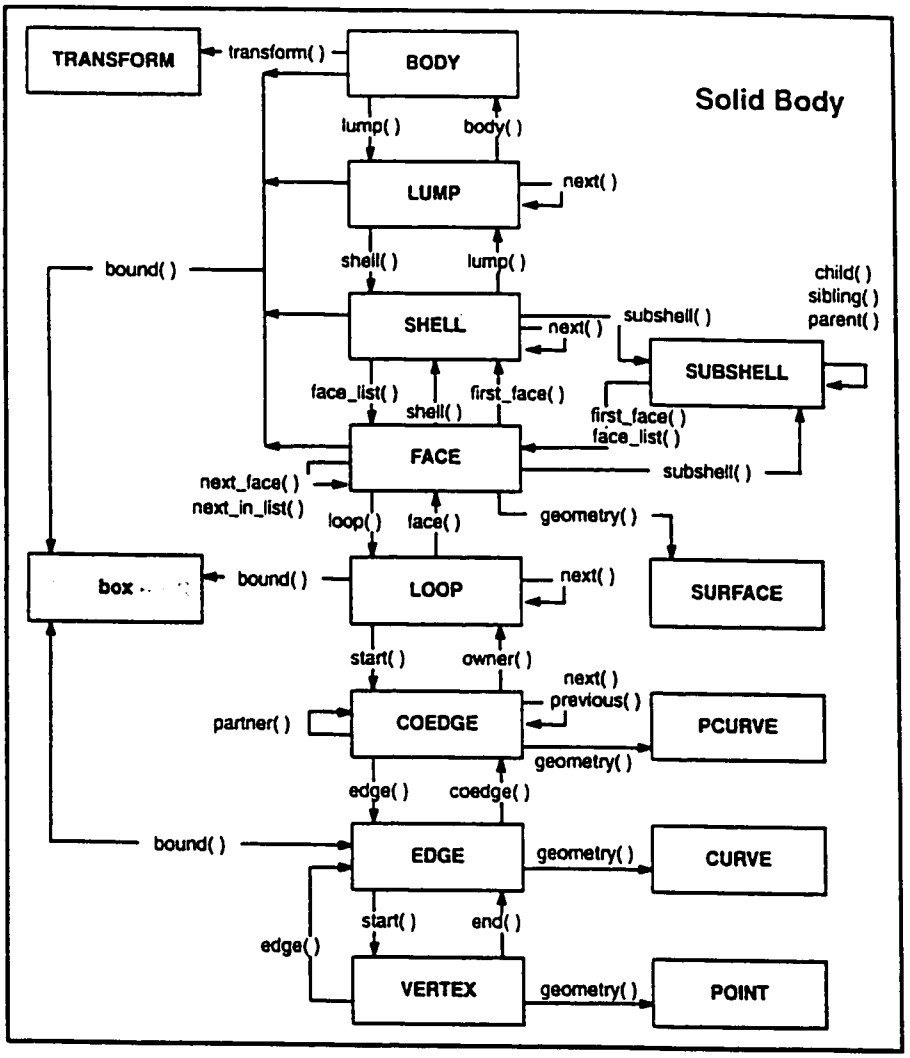


Figure 2.8: Implementation of solid BODY [51]

2.1.4 Free-form surface representation

Non-Uniform Rational B-Spline (NURBS) representations of free-form curves and surfaces have achieved widespread acceptance in the CAD/CAM applications [22][35] [30][47][79]. The main reasons can be summarized as follows: 1) NURBS are invariant under scaling, rotation, translation, and projections; 2) They offer common mathematical form for representing analytic and free-form curves and surfaces; 3) Evaluations are fast and stable; and 4) A wealth of geometric algorithms (a geometric toolkit) is available for designing, analyzing, processing, and interrogating the objects.

A NURBS curve is a piecewise rational polynomial vector function with the form of ([22][47])

$$\mathbf{C}(u) = \frac{\sum_{i=0}^{n} w_i \mathbf{P}_i \mathbf{N}_{i,p}(u)}{\sum_{i=0}^{n} w_i \mathbf{N}_{i,p}(u)} \tag{2.1}$$

where w_i are so-called weights, $\mathbf{P}_i$ form the control polygon approximating the curve, and $\mathbf{N}_{i,p}(u)$ are the normalized B-spline basis functions of degree p defined as follows.

$$\mathbf{N}_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{N}_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i}\mathbf{N}_{i,p}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}}\mathbf{N}_{i+1,p-1}(u) \tag{2.2}$$

u_i are the knots forming the knot vector

$$\mathbf{U} = \{\mathbf{u_0}, \mathbf{u_1}, ..., \mathbf{u_m}\} \tag{2.3}$$

where m is related to the degree p and the number of control points n by $m = n+p+1$.

The tensor product generalization of the NURBS curve defines a NURBS surface as follows ([22][47]).

$$\mathbf{S}(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m} w_{i,j}\mathbf{P}_{i,j}\mathbf{N}_{i,p}(u)\mathbf{N}_{j,q}(v)}{\sum_{i=0}^{n}\sum_{j=0}^{m} w_{i,j}\mathbf{N}_{i,p}(u)\mathbf{N}_{j,q}(v)} \tag{2.4}$$

where $w_{i,j}$are again weights, and $\mathbf{P}_{i,j}$ form a control net approximating the surface. $\mathbf{N}_{i,p}(u)$ and $\mathbf{N}_{j,q}(v)$ are normalized B-spline basis functions of degree p and q, respectively. u and v are defined over the knot vectors:

$$\mathbf{U} = \{\mathbf{u_0}, \mathbf{u_1}, ..., \mathbf{u_{m_u}}\} \tag{2.5}$$

$$\mathbf{V} = \{\mathbf{v_0}, \mathbf{v_1}, ..., \mathbf{v_{m_v}}\} \tag{2.6}$$

In the CAD/CAM applications, the most commonly used surfaces are swept surfaces, skinned surfaces, extruded surfaces, surfaces of revolution, Ruled and Coon surfaces [79]. Roughly speaking, the main idea in constructing these surfaces is generate a NURBS surface which interpolates a set of curves. For example, a general form of the swept surface is given by

$$\mathbf{s}(u,v) = \mathbf{T}(v) + \mathbf{A}(v)\mathbf{C}(u) \tag{2.7}$$

where $\mathbf{C}(u)$ and $\mathbf{T}(v)$ denote a section curve and a path curve, respectively. $\mathbf{A}(v)$ is the general transformation matrix to translate and orient the section curve along the path curve.

2.2 Geometric modeling of milling operations

Relevant published research work in the geometric modeling of milling processes can be summarized as follows. Bertok et al [12] proposed a torque prediction system based on the geometric simulation of the flat-end milling operation. Their geometric model was able to compute the removed volume per tooth for simple geometries. The swept volume of solids of revolution was investigated by Wang [69][70][71] and Wang and Wang [72]. Based on the equation of surfaces representing the boundary of the solid and the motion function (feed), a family of curves lying on the envelope of the moving surface was generated. The authors then constructed the 3D pixel space swept volume which was used for subsequent Boolean operations. However, their spatial partitioning approach is view-dependent (i.e. in the case of a change in the eye-view, the entire representation must be reconstructed which limits its applications). Huang and Oliver [31] also used the spatial partitioning approach to predict NC milling errors. Instances of tool positioned along the tool path are consecutively used to update the part by one-dimensional regularized Boolean operations. They overcome the view dependency problem by introducing a contour display method. To accelerate the machining simulation and verification, Menon and Robinson [41] enhanced the work by Menon and Voelcker [42] with parallel computation. They also extended the range of verification to the machining tolerance and static deflection

assessments. Although, the above spatial partitioning approaches degrade much more slowly with the complexity of the updated workpiece as opposed to the B-rep, and it is relatively easy to implement parallel spatial partitioning, these methods generate an approximated representation of 3D solids. To increase the level of precision, a large amount of memory is required.

Based on Constructive Solid Geometry (CSG) part representation, Spence [53][2] performed computer simulations of flat-end milling operations. The instantaneous immersion geometry (entry and exit angles) was computed using a semi-circle sweeping the tool path. Ball-end milling simulation was reported in El-Mounayri et al [19] using a B-rep polyhedral-based solid modeler. The removed volume was computed and the cutting edge, modeled with a cubic Bezier curve, was intersected with the volume to find the in-cut segments. Studies for 3-axis chip geometry calculation are limited to Feng and Menq [24]. They used a commercial CAD/CAM software (CATIA), to determine the boundary of the engaged surface between the tool and the workpiece. The in-cut segments were then computed using the boundary curves. However, the geometric calculations were not based on an information-ally complete model of the in-process part.

2.2.1 Milling geometric simulation requirements

In order to simulate the milling operations of the sculptured surfaces, the geometric modeling approach must support an accurate, unambiguous, and valid model of the part. Also, it should be possible to efficiently update the model throughout different stages of the simulation. Lastly, the geometric modeling approach must efficiently construct entities needed in the machining simulation such as cutting edges and the

chip geometry.

B-rep solid modeling techniques offer generality, accuracy and completeness in 3D part representation [35][30]. This makes them appropriate for the geometric simulation of machining processes. An accurate and consistent model of the part is available throughout different stages of the simulation. Also, it is possible to construct the accurate representation of entities needed in the machining simulation.

The bottleneck of most available B-rep modelers is that the Boolean operation is not reliable [35]. B-rep modelers can reliably handle analytical forms where the closed-form (or exact) representations of entities, such as surface-surface intersections are available. However, when they are dealing with free-form surfaces, the exact representation of entities is not possible. Therefore, an approximated piecewise entity fitted to points on the exact geometry is constructed [43]. In general, the main reason for the failure of applications such as part updating, chip geometry calculation, and also feed-mark construction in machining simulation, is large approximation (or fitting) tolerances.

To maintain the advantages of B-rep techniques and overcome the limitation, the current research proposes an improved swept volume and surface generation method. In the developed method, boundary faces of the swept volume are constructed using advanced sweeping/skinning techniques [47]. In addition to position data, derivative constraints are imposed in order to improve approximation tolerances.

2.3 Physical simulation of milling operations

2.3.1 Milling force model

There has been significant research work reported in modeling the mechanics of flat-end milling. Average milling forces have been related to important process variables such as feed, depth of cut (DOC), and width of cut (WOC) by means of empirical equations [4]. Also, semi-empirical or mechanistic force models have been introduced to predict the instantaneous cutting forces as well as the average power and torque [62][33][75][77].

An alternative approach to the mechanistic force model is the analytical or fundamental force model which is based on the mechanics of the cutting operation on the shear plane and the cutting edge [5][14]. The main idea in this approach is to relate the cutting coefficients to the basic cutting quantities (i.e., the shear stress, shear angle, and friction angle) found from orthogonal cutting tests. Recently, Budak et al [15] reported an orthogonal based method which also includes the edge forces (components of cutting force which are independent of chip thickness) to predict three components of flat-end milling forces. This orthogonal based approach has been applied to various machining operations such as drilling, turning and milling by Armarego et al [9][8][6][7][5]. However, this approach to the flat-end milling (or ball-end milling) suffers from the fact that it highly depends on the accuracy of the orthogonal cutting data. In addition, the issue of predicting the chip flow angle for flat-end milling needs to be investigated further.

Models reported for ball-end milling force prediction are limited to Yang and Park [74], Abrari and Elbestawi [1], Feng and Meng [23][25] and Yücesan and Altintaş

[78]. Yang and Park [74] introduced a two-component force model which predicts the empirical cutting force coefficients from a set of orthogonal turning tests, based upon the theory of equivalent orthogonal conditions. This model was experimentally validated for light immersions only. Also, the effect of edge forces was not included in their model. Abrari and Elbestawi [1] have developed a closed form force model for flat- and ball-end mills. In this model, a set of closed form functions are introduced for the projected areas of chip load on the reference coordinate planes. A pair of cutting forces, normal and tangential, are associated with each projected area. An average specific pressure matrix relates the projected areas to two components of cutting forces. Feng and Meng [23][25] introduced a two-component rigid force model that considers local cutting mechanics of differential cutting edges by introducing K_t and K_r as polynomials of the axial depth of cut. The size effect in metal cutting was also included in their model. However, cutting edge geometry (such as helix and rake angles) and edge forces were not included in this model. Yücesan and Altintaş [78] considered the normal and friction forces on both rake and clearance faces. They assumed that the loads on the rake face are proportional to the uncut chip thickness area and the loads on the flank face are concentrated on the in-cut portion of the cutting edge. They suggested that the variations of normal and friction coefficients depend on the cutter, immersion geometry, and feed rate. However, they assumed constant (average) values for chip flow angle and cutting coefficients along the cutting edge. Calibration and verification of their model was also limited to simple two-dimensional cases only (i.e., slotting and half-immersion) which have very few practical applications.

2.3.2 Machine tool dynamics

The vibratory system of the tool and machine structure in end-milling operations has been studied by Tlusty et al [63][59][60][61]. Typically, this vibratory system has two dominant modes [61]: 1) The vibration mode of the end-mill which has higher natural frequency as well as higher flexibility; and 2) The vibration mode of the spindle with bearings acting as a spring. A lumped mass-damper-spring model is proven to be adequate for simulating the tool-structure dynamics [63]. Two degrees of freedom are considered in each of two non-rotating perpendicular directions. The dynamic characteristic are measured by performing modal testing on the machine tool and spindle-tool unit.

2.3.3 Closed loop of machine tool-cutting process

In the milling operation, the machine tool structure and the cutting process can be regarded as two distinct elements which form a closed loop [34]. The input to the machine structure is the cutting forces produced by the cutting action at the cutting edge(s)-tool interface. The outputs of the machine tool structure are the deflections which, in turn, affect the undeformed chip geometry. In order to physically simulate the milling operation in addition to the machine tool and cutting process transfer functions, their interactions must be taken into consideration.

The research work reported on integrating the transfer functions of the machine tool and the cutting process are reviewed by Smith and Tlusty [50]. One important mechanism affecting the machine tool-cutting process closed loop is the regeneration [34]. The chip load on the cutting edge depends not only on the feed and the current

deflection of the tool, but also on the surface which was cut by the preceding cutting edge(s). This mechanism is known as the regeneration. As the amplitude of vibrations grows, the cutting edge can jump out of the cut which results in zero forces. This phenomenon is known as the basic non-linearity in the cutting process [60]. The other mechanism affecting the cutting process is the ploughing mechanism which occurs in the vicinity of the edge nose. The ploughing force is of interest not only in connection with the analysis of the balance of forces but also in explaining the amount of damping generated in the cutting process. Different approaches have been used to model the ploughing forces (for example, see Elbestawi et al [20]).

2.4 Simulation systems

Due to the complex tool and chip geometry, comprehensive geometric and physical simulations of milling operations are rather involved tasks. Although, there has been some research work (reviewed in sections 2.2 and 2.3) in either geometric modeling or the physical modeling of the process, little work has been performed on the integration of the two modeling environments.

Takata et al [58] developed a system composed of geometric simulation and milling mechanics to estimate the machining errors generated by the tool deflection. The geometric simulation was basically an extended z-buffer technique used by Wang and Wang [72]. The physical simulation further improved in their later work [65] to predict the onset of chatter vibrations. Also, they implemented the system [66] to automatically generate optimum tool paths. The objective was to maximize the metal removal rate (MRR) subjected to stable machining and accuracy constraints. Takata

[57] proposed a machining scenario to represent the simulated machining process. Two major applications for the scenario were suggested: 1) On line control of the machining process and 2) Monitoring and diagnosis of machining operations.

Based on the CSG part representation, Spence et al [53] computed the cutter/part immersion geometry. They used computational geometry methods to efficiently find the relevant geometry to the milling simulation [55]. Also, they integrated the immersion geometry module with a mechanistic force model [2] in order to: 1) Schedule the feed subjected to the maximum force and machining error constraints, and 2) Assist the adaptive controller to decrease the force overshoot [54].

In the reviewed literature, only $2\frac{1}{2}$-axis flat-end milling operations for $2\frac{1}{2}$D pockets with simple geometry were taken into consideration. For these operations, the chip geometry extractions are rather simple. For example, when the DOC is constant the only parameters that need to be computed are entry and exit angles. This assumption greatly simplifies the chip geometry extraction which is reduced to the semi-circle/curve intersection calculation. This has been due to the lack of comprehensive geometric modeling techniques which can simulate the complex geometry involved in three- to five-axis milling operations of the sculptured surfaces. The accuracy and applicability of the physical models used are also an important issue, particularly for complex cutting edge geometries. Moreover, the integrated simulation system must be developed for sculptured surface milling operations subjected to a variety of cutting edge geometries, cutting conditions, and immersion geometries.

Chapter 3

Model-based milling simulation methodology

3.1 Basic concept

The comprehensive and realistic simulation of the milling operations requires two modeling sub-environments: 1) A 3D information-ally complete geometric modeling environment which can support a variety of mathematical surfaces (including NURBS) as well as a library of geometric algorithms; 2) A realistic simulation environment for modeling physical/mechanical properties of the milling operations with equivalent mathematical models. Another critical requirement is the realistic simulation of the interactions between the geometric model and the physical/mechanical model. To achieve comprehensive and efficient milling simulation, the above requirements must be integrated in order to form the core of the simulation environment (Figure 3.1).

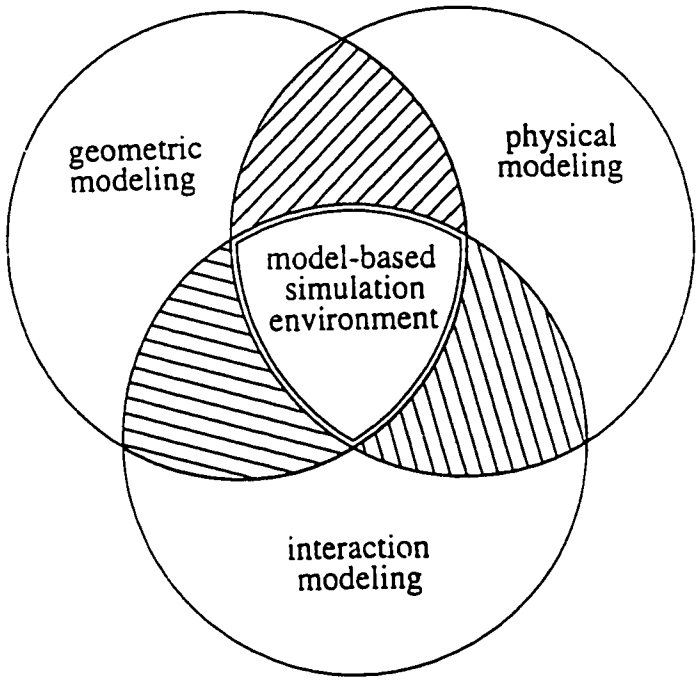


Figure 3.1: Model-based simulation environment

Depending on the application, the model-based simulation environment can be developed with varying levels of accuracy and complexity. For example, the geometric modeling can be implemented to update the part either by the simplified swept volume of the cutter (assuming the cutter as a solid of revolution) or by the true surface swept by the cutting edges. For the roughing and semi-finishing operations (Chapter 4 and Chapter 5), the first approach is developed and implemented. However, for the surface texture analysis (Chapter 6) the second approach must be implemented.

3.2 Geometric modeling

As mentioned in Chapter 2, solid modeling techniques provide a reliable model-based environment for 3D part presentation. This makes them appropriate for the milling

process simulation where an information-ally complete representation of entities such as the part, tools, and swept volume is fundamental.

In solid modeling, one of the most widely used methods for 3D part representation is Boundary Representation (B-rep) for the following reasons. B-rep can support a variety of mathematical surfaces including the most versatile one (i.e., NURBS). Also, in the B-rep data structure, faces and edges are readily available together with a wealth of algorithms and functions, to either extract required geometric information or modify the 3D model of workpiece. For example, the swept objects encountered in CAM applications are effectively modeled in the B-rep.

In this research STI's ACIS [52], a 3D B-rep geometric kernel, is used as a geometric engine to simulate the milling process. ACIS can support the NURBS as well as analytical surfaces. The new applications can interact with ACIS in C++, Scheme or a combination of both.

The B-rep geometric modeling approach proposed here involves the following procedures: 1) Generating the B-rep model of the stock; 2) Generating the B-rep model of the design part; 3) Representing different entities encountered in the milling simulation (for example, cutting edges, rake faces, and so forth); and 4) Updating the B-rep model by performing Boolean operations. The last procedure requires the swept volume(s) of the cutter generated by the interaction modeling. Different levels of accuracy for the geometric modeling are discussed and implemented in the following chapters.

3.3 Physical modeling

In the milling operation, the machine tool structure and the cutting process can be regarded as two distinct elements which form a closed loop (Figure 3.2). The inputs to the machine structure are the cutting forces produced by the cutting action at the cutting edge(s)-tool interface. The outputs of the machine tool structure are the deflections which affect the undeformed chip geometry. In order to physically simulate the milling operation, the machine tool transfer function and the cutting process transfer function are required. In the following chapters the machine tool transfer function and the cutting process transfer functions are developed and implemented.

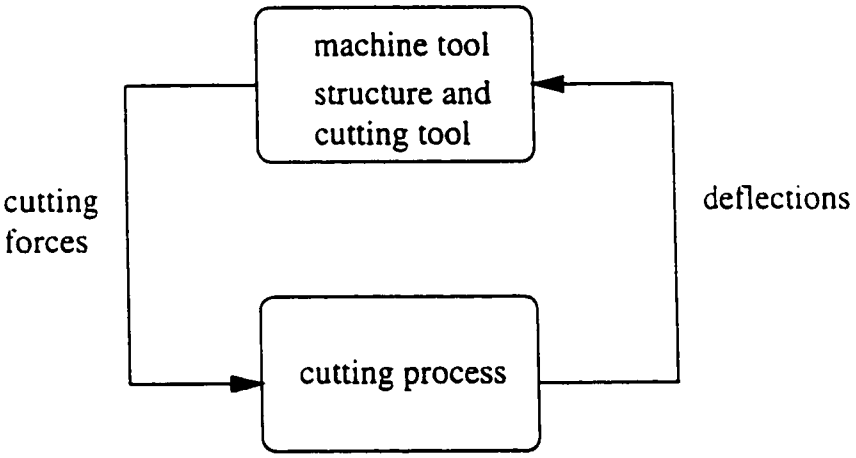


Figure 3.2: Closed loop of the machining unit

3.4 Interaction modeling

The B-rep model of the part is constantly varying as the milling cutter removes the material. The first step in modeling the interactions is to generate the swept volume of the cutter (or the swept surface of the cutting edges). The swept object is passed to the geometric modeling module for part updating. The second step is to construct a model of the undeformed chip geometry. In the literature reviewed in Chapter 2 analytical relationships were developed to computed the chip geometry for flat- and ball-end mills. However, they were limited to simple geometries. In this research the chip geometry is constructed for three-axis milling of sculptured surfaces (Figure 3.3) using B-rep solid modeling techniques.

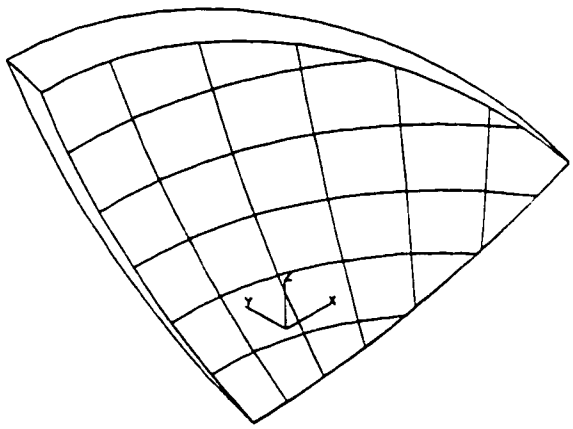


Figure 3.3: B-rep model of chip in ball-end milling

The tool deflections under the cutting forces affect the updated part geometry. In some cases the tool deflections can cause surface gouging problems. In the swept volume generation method proposed here, the tool deflections can also be taken into consideration.

Chapter 4

Two-axis roughing operation

This chapter deals with a model-based simulation methodology for die cavity roughing operations. In the first section tool path patterns used for two-axis roughing (pocketing) operations are reviewed. The optimum pattern based on the objective of maximum MRR subjected to the stability and tool tip deflection constraints is proposed. The second section deals with the geometric simulation and the undeformed chip geometry extraction for the pocketing operation. In the next section, first the research works relevant to the physical simulation of the roughing operation are reviewed. Then, the module is implemented to predict cutting forces, tool deflections, and the onset of chatter vibrations. The simulation and experimental results are then compared. The application of the methodology in feed scheduling subjected to the dominant constraints is discussed in the next section. The last section summarizes the roughing simulation methodology developed.

4.1 Optimum tool path pattern

In general, patterns used for flat-end milling can be categorized as follows: 1) one-way pattern; 2) zig-zag pattern; 3) spiral-in; and 4) spiral-out patterns (for example, see [56]). Also, for the roughing operation the offset approach with ball-end mills has been used [67][17]. However, this type of cutter is not as efficient as the flat-end mill for the stock removal [68]. Dragomatz and Mann [18] published a classified literature review on the NC milling tool path generation techniques. The reviewed papers were drawn from the engineering, computer science, and mathematics disciplines.

Zig-zag and one-way patterns have been preferred for a long time over spiral strategy mainly due to the fact that the offset computation has been considered time consuming and prone to errors. Persson [45] initiated a technique to spirally machine the pockets with linear and circular boundaries. In his approach offsets were efficiently achieved using the bisector (or Voronoi) diagram (Figure 4.1). Persson's approach was modified by Held et al [28][29] to include the orientation on the bisectors according to increasing distance, (Figure 4.2). Also, pockets with bottlenecks are partitioned into sub-areas which contain just one innermost point. Held also discussed the issues related to the technological optimization. For example, the upper bound of tool diameter was selected based on the minimum convex corner radius and the minimum width of bottlenecks of the pocket. In addition, the maximum, minimum, and average width of cut for the next move are extracted. However, the variations of chip geometry and its relation to cutting forces are not taken into consideration. The research performed by Ferreira and Hinduja [26] discusses another method of generating the spiral tool paths. Using the boundary shrinking method, the offsets are computed

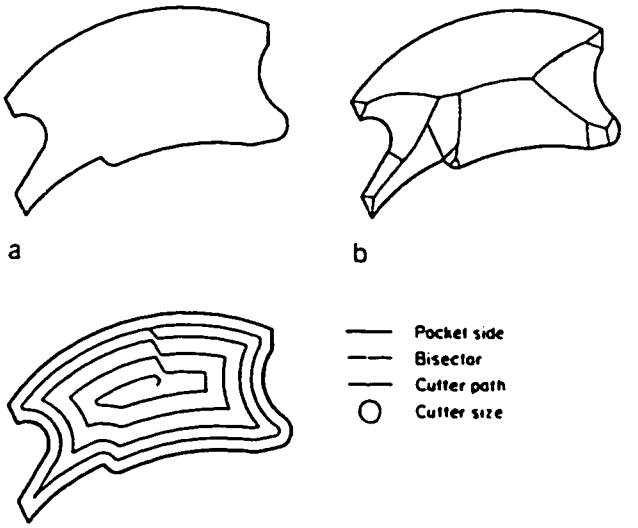


Figure 4.1: Pocket machining and bisector diagram[45]

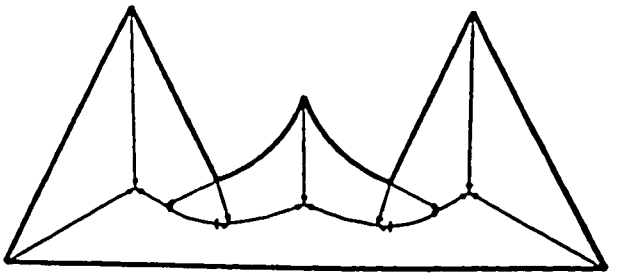
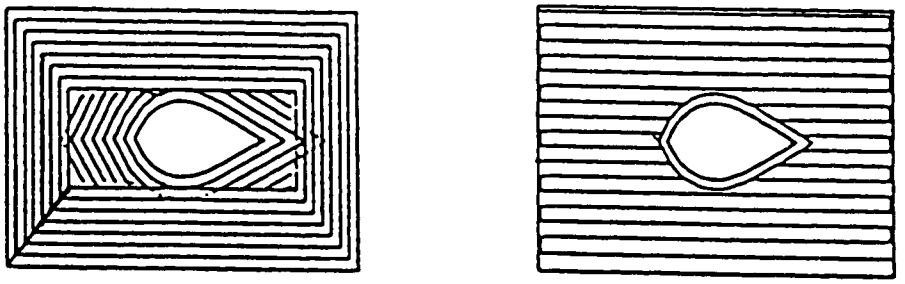


Figure 4.2: Bisector diagram enhanced with orientations [28]

using the tool diameter and the WOC(a user-input). They also tried to maximize the length of tool path over which down-milling takes place. To increase MRR, they subdivided each path into constant WOC portions and selected the feed based on the WOC.

The issues related to ship hull machining were investigated by Vickers et al [67][17]. Among six 2D tool path patterns, they reported that parallel offset and stock-component offset patterns, (Figure 4.3), are the most efficient ones. They also tried to minimize the total machining time subjected to some physical constraints, such as the spindle power and the spindle torque. Tool paths computed using a sta-



a- stock/component offset pattern b- parallel offset pattern

Figure 4.3: Efficient tool path patterns [17]

bility data bank were investigated by Tlusty et al[64] and Weck et al[73]. In the latter work, the cutter starts in the slotting mode to machine the pocket boundary. The stable DOC is extracted from the stability data bank. Then, the inside material is removed using one-way pattern in down-milling which was reported to give the shortest machining time.

The automatic and error-free generation of tool paths were the primary concern of the above researchers. For example, geometric aspects such as the tool diameter,

constant WOC segments, and similar issues were addressed. Also, the issues related to the technological optimization were considered which were restricted to conservative cutting conditions selected from machining data handbooks or rule of thumb. However, the comprehensive technological optimization depends on the fundamental constraints of machining process, which are chatter vibrations and cutter deflections.

Proposed tool path pattern

The optimum tool path pattern for 2D pocketing operation considering main constraints of chatter vibrations and tool deflections is designed as follows. The one-way pattern is considered for removing the inside material. The direction of the one-way pattern is positioned along the most stable direction of the machining tool. Then, a contour tool path removes the uncut material left on the boundary of pocket (Figure 4.4).

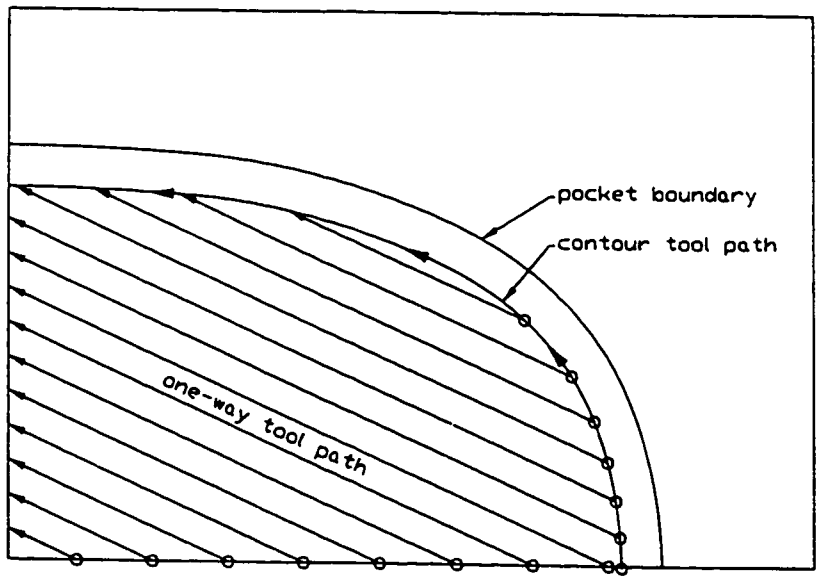


Figure 4.4: Proposed tool path for pocketing operation

For the following advantages, the proposed pattern can substantially decrease the

total machining time:

1. Positioning the one-way direction along the most stable direction of the machine tool substantially increases the MRR [64][73].

2. Using this pattern the cutting operation is always performed in the down-milling mode. The stability and the tool life are higher in this mode [64][26].

3. The cornering problem [66] is never encountered when removing inside material.

4. The tool deflection is not a constraint in removing inside material. Thus, MRR can go beyond this limit.

4.2 Geometric simulation of flat-end milling

This section deals with issues involved in the geometric modeling of the roughing operation. First, the volume removed by the cutting edge is modeled by the volume swept by two-axis translatory motions of the solid of revolution (the cutter). Two types of swept volumes are considered in this section: 1) the swept volume generated for inside paths ignoring the tool deflections, and 2) the swept volume generated for the contour path considering the tool deflections. The roughing operations are then simulated by performing consecutive Boolean subtractions between the B-rep model of the part and the cutter swept volume. Finally, the undeformed chip geometry is extracted from the updated part model.

4.2.1 Two-axis swept volume generation

To precisely simulate the material removal of milling operation, the removed volume must be constructed based on the true path of each point on the cutter edge (Figure 4.5). However, for non-functional surfaces the true swept volume can be simplified to the volume generated by the translational sweeping motion of a solid of revolution, as proposed by Wang et al [72] and Menon et al [42]. For flat-end milling operations, the solid of revolution (henceforth referred to as the cutter) is basically a cylinder with the same diameter as the tool and with the same height as the total height of the cutting edges as depicted in Figure 4.6.

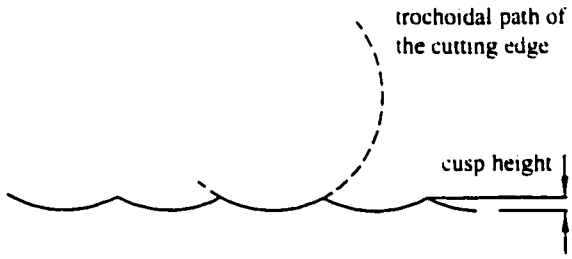


Figure 4.5: Feed marks or cusps left on machined surface

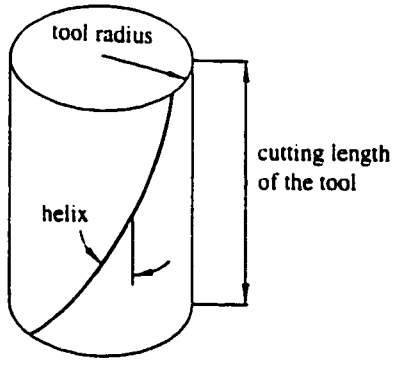


Figure 4.6: Flat-end mill geometry

Two-axis swept volume: rigid tool

With the assumption of the rigid tool, two-axis swept volume generation is straightforward. The geometric kernel used provides us with the function for sweeping operations along a given 2D path. The tool profile and the tool path are the required inputs for the function. After generating the path-dependent part of the swept volume, portions of the cutter at initial and final positions are added using the Boolean unite operation (Figure 4.7).

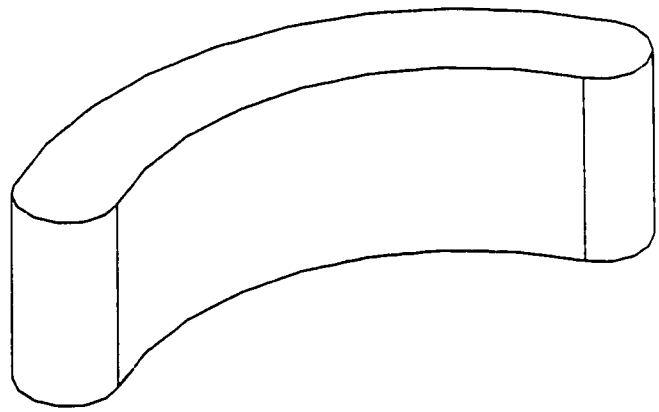


Figure 4.7: two-axis swept volume: rigid tool

Although, the edges of the tool profile are analytic (for this case, simple linear edges), the geometric kernel generates NURBS surfaces while sweeping the tool profile. In this regard, the fitting tolerances may cause serious problems during the updating process. Further details as well as a new sweeping/skinning technique can be found in Appendix A and Section 5.2.1.

Two-axis swept volume: flexible tool

The swept volume can be constructed by assuming that the vertical edges of the tool profile have the same shape as the tool axis deflection curve. The tool axis deflection curve is constructed based on the the maximum tool tip deflection, δ_x and δ_y, which are computed in Section 4.5. The deflection curve of a cantilever beam with a concentrated load applied at the end point can be expressed by

$$\delta_z = \frac{Pz^3}{6EI} - \frac{Pl^2z}{2EI} + \frac{Pl^3}{3EI} \tag{4.1}$$

Knowing the maximum deflection(δ_{max}) and its direction ($\theta_{\delta_{max}}$) at the tool tip, the above relationship can be rewritten as follows:

$$\delta_z = \frac{\delta_{max}}{2}(x/l)^3 - \frac{3\delta_{max}}{2}(x/l) + \delta_{max} \tag{4.2}$$

$$\delta_{max} = \sqrt{{\delta_x}^2 + {\delta_y}^2}$$

$$\theta_{\delta_{max}} = \arctan \frac{\delta_y}{\delta_x}$$

Thus, the tool profile deflected under cutting forces can be represented by, as depicted in Figure 4.8, two B-spline edges (AC and BD) fitted to the deflection curves on the edges together with two linear edges (AB and CD) representing top and bottom faces of the cutter.

Instances of the deflected tool profile are constructed along the contour tool path knowing the maximum deflections (i.e., δ_x and δ_y) at each tool position (Figure 4.9). The advanced sweeping/skinning techniques which will be discussed in Section 5.2.1 are then used to generate the faces corresponding to each edge of the tool profile. The

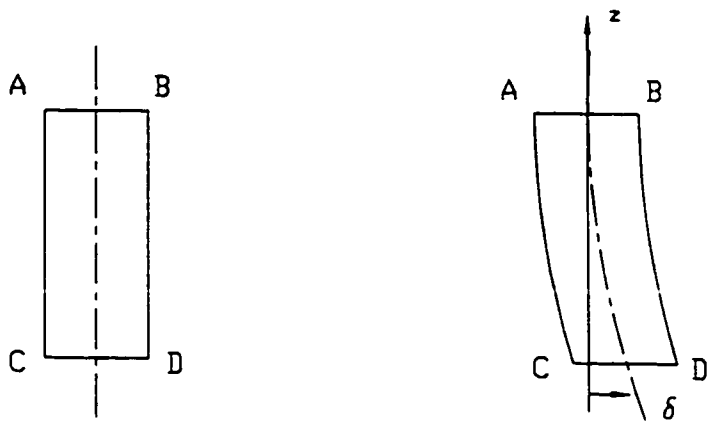


Figure 4.8: Deflection of tool profile

faces representing portions of the cutter at initial and end points are also constructed. Lastly, the ACIS stitching operation is used to generated a topologically valid swept volume (Figure 4.10).

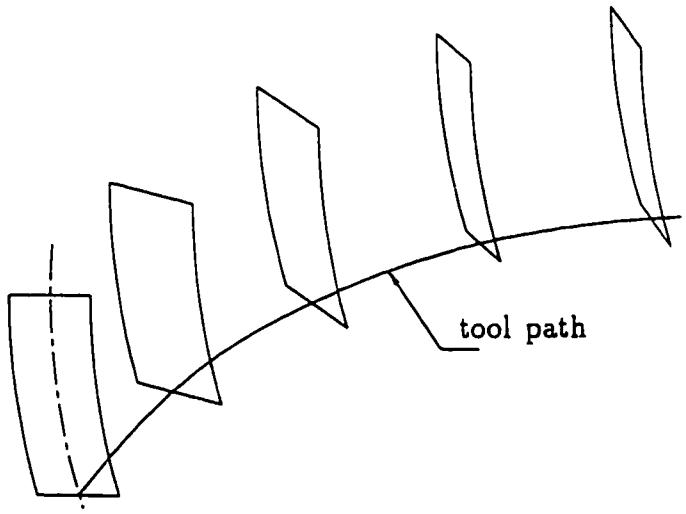


Figure 4.9: Instances of tool profile along two-axis tool path

4.2.2 Part updating in roughing operation

In the part updating process, a B-rep model of the stock is constructed which is updated throughout roughing, semi-finishing, and finishing operations. For presentation

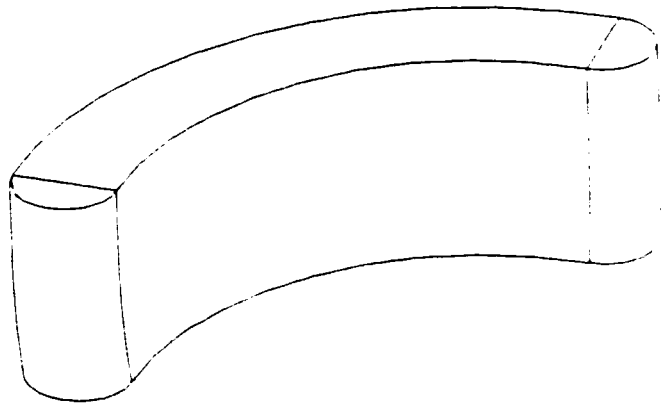


Figure 4.10: Two-axis swept volume: flexible tool

purposes, one quarter of a die with sculptured surfaces is considered (Figure 4.11). In the next step, the inside material is removed by linear-path swept volumes. The swept volumes of inside tool paths do not generate the die surface; thus, the tool deflection is ignored in generating swept volumes. However, the swept volume of the contour path is close to the die design surface. Therefore, the tool deflections must be taken into consideration. A B-spline path representing a few blocks of the CL data file is used to generate the swept volume. The number of blocks depends on the path curvature. In the last step, Boolean subtractions are performed which consecutively updates the part (Figure 4.12). Thus, it is possible to observe the precise simulation of the machining operations step-by-step and, in case of any error, corrective actions can be taken before actual machining.

The B-rep model of the rough-cut part can assist the tool path planning of the following operation, that is, the semi-finishing. For this operation, the tool paths can be computed based on the knowledge of the material remaining on the design surface and precisely represented by the B-rep model (Figure 4.13). The B-rep model of the remaining material includes shoulders (or steps) left by flat-end milling as well as

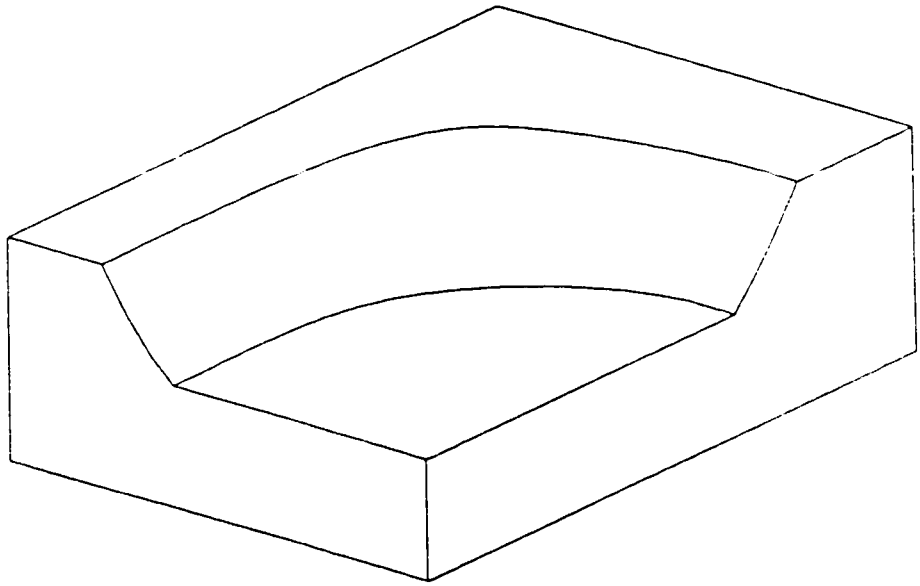


Figure 4.11: One quarter of die with sculptured surfaces

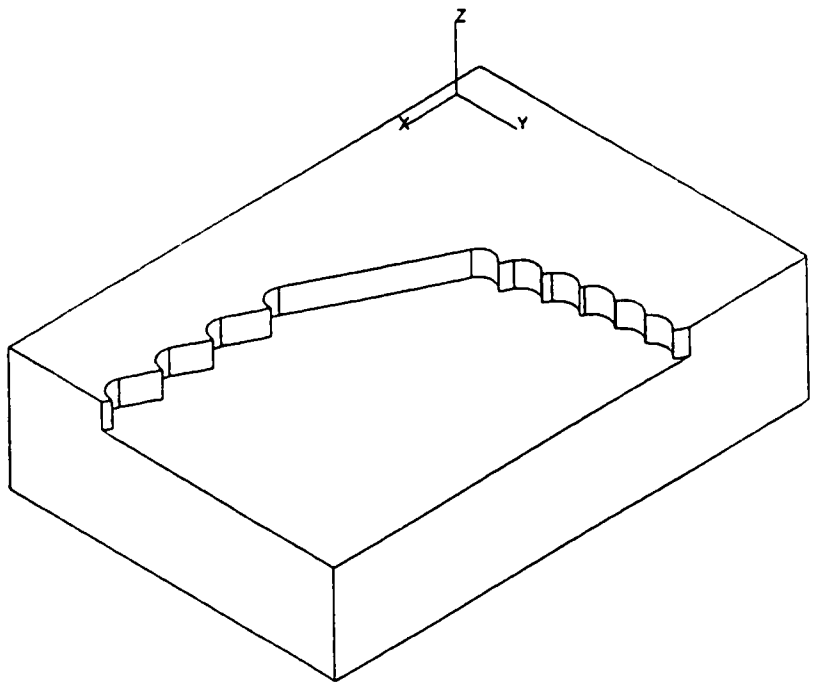


Figure 4.12: Updated part for the first one-way pass

the material left as a result of the tool normal-to-path deflection. Also, the gouging regions due the tool tangent-to-path deflection can be identified. This issue will be dealt with in Section 5.5.

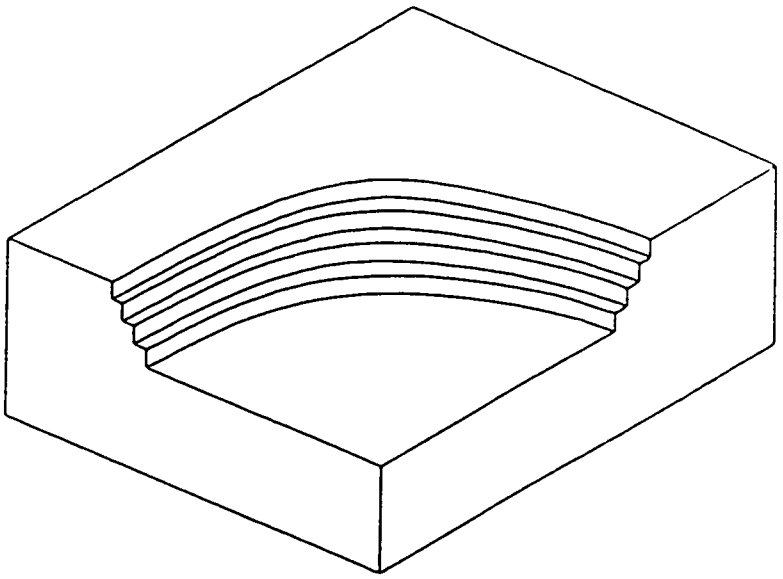


Figure 4.13: B-rep model of rough-cut part

4.2.3 Instantaneous undeformed chip geometry

Accurate and reliable chip geometry can be extracted from the B-rep model of the part. The B-rep model is updated as the actual cutter removes the material. In addition to the updated part, the chip geometry for flat-end milling depends on the cutting edge design, the tool path, and the feed. The cutting edge design for constant helix flat-end mills is determined by the tool radius R and the helix angle β (Figure 4.6).

Undeformed chip thickness

The instantaneous undeformed chip thickness is one of the most important parameters influencing cutting force[62]. Martellotti [40] derived the following equation for the chip thickness by assuming a circular path for the motion of the points on the cutting edge

$$t_c(\theta) = R + f_t \sin\theta - \sqrt{R^2 - {f_t}^2 \cos^2\theta} \tag{4.3}$$

where R, f_t, and θ are the tool radius, feed per tooth and edge angular position, respectively. For practical face milling operations, the above relationship was further simplified to

$$t_c(\theta) \approx f_t \sin\theta \tag{4.4}$$

The above relationship has been used to compute the chip thickness in the flat-end milling operation.

Incut segments

Knowing the edge geometry (the tool radius and helix angle), the problem of computing the entry and exit points on the cutting edge is simplified to computing the intersection points of a semi-circle with the B-rep model of the part. The center of the semi-circle is placed at the tool position on the tool path. Also, as shown in Figure 4.14, the semi-circle is oriented in such a manner that its axis is aligned with the tool path tangent.

Instances of the semi-circle are swept along the tool path and the intersection points are reported. Figure 4.15 shows the incut segments of the semi-circle along the contour path. The entry and exit angles are then computed using the intersection

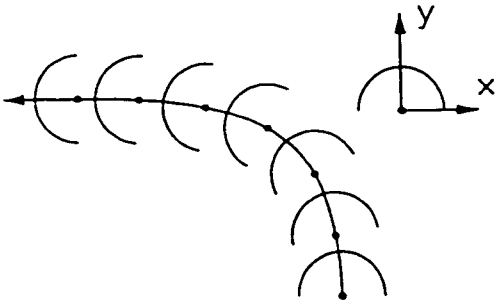


Figure 4.14: Oriented semi-circles along the tool path

points. The variation of immersion geometry along the tool path is illustrated in Figure 4.16. The extracted immersion geometry is a fundamental parameter for the accurate prediction of the cutting forces.

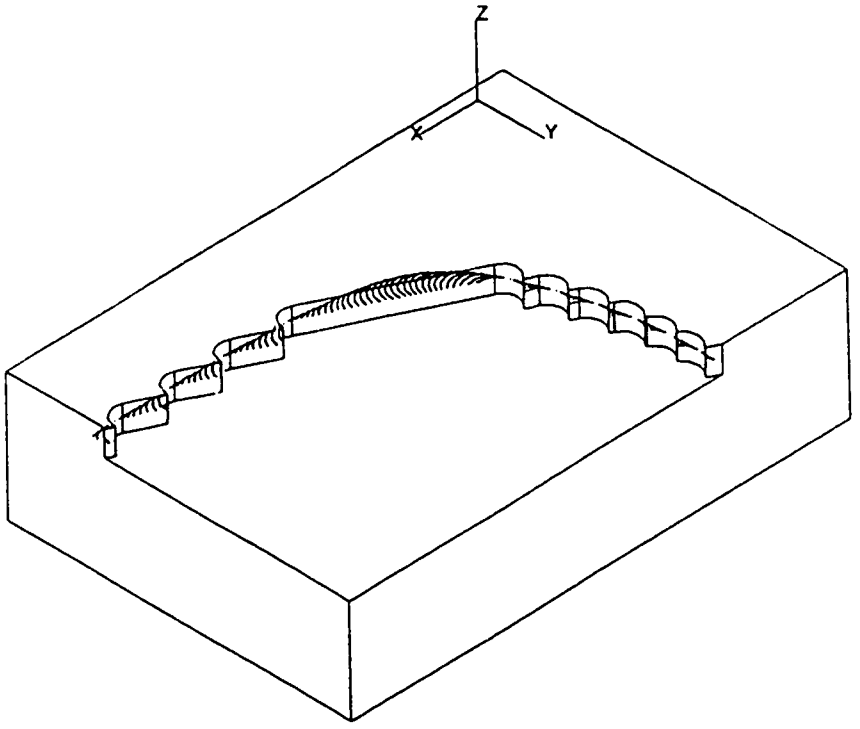


Figure 4.15: Incut segments of semi-circles along the tool path

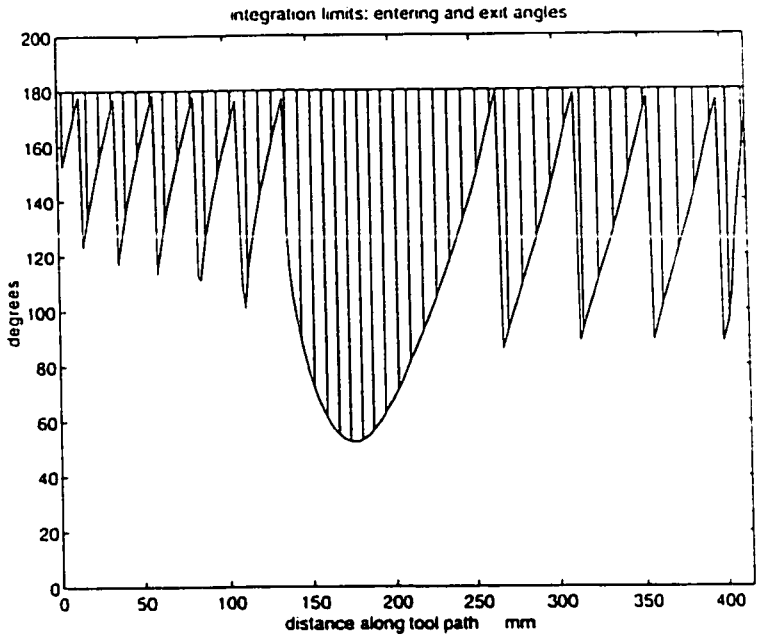


Figure 4.16: Variation of immersion geometry along the tool path

4.3 Physical simulation of flat-end milling

There has been quite significant research work reported in developing the transfer functions of machining process (Figure 3.2) with varying levels of accuracy and complexity. A review of these works was presented by Smith et al [50]. In this section, the machine tool transfer function as well as the cutting process transfer function for flat-end milling are proposed.

4.3.1 Force model for flat-end milling

In the current work, the mechanistic force model proposed by Tlusty and MacNeil [62] is implemented for the flat-end milling operation. In this model, it is assumed that the differential tangential force dF_t is proportional to the differential chip load,

and the differential radial force is proportional to the tangential force:

$$\begin{aligned} dF_t &= K t_c dz \\ dF_r &= c dF_t \end{aligned} \tag{4.5}$$

where K is a cutting coefficient which is mostly determined by the material, the average chip thickness, and the tool geometry. The tangential and radial forces, dF_t and dF_r, when applied at the center of the cutter can be decomposed into the xy non-rotating coordinate system (Figure 4.17):

$$\begin{aligned} dFt &= -dF_t \sin(\phi) + dF_r \cos(\phi) \\ dFt &= -dF_t \cos(\phi) + dF_r \sin(\phi) \end{aligned} \tag{4.6}$$

where ϕ is the angular position of the element on the cutting edge. To integrate the above equations at the cutting edge angular position ϕ_j, the integration limits (or the end points of the incut segments of the cutting edge) are needed which are provided by the geometric simulation module. Thus, the total predicted cutting forces, F_x and F_y, at the angular position ϕ_j can be expressed by

$$\begin{aligned} F_x(\phi_j) &= \sum_{k=1}^{Z} \int_{\phi_{in}}^{\phi_{out}} dF_x \\ F_y(\phi_j) &= \sum_{k=1}^{Z} \int_{\phi_{in}}^{\phi_{out}} dF_y \end{aligned} \tag{4.7}$$

where ϕ_{in} and ϕ_{out} are the entry and exit angles of the cutting edges and Z is the number of edges involved in cutting.

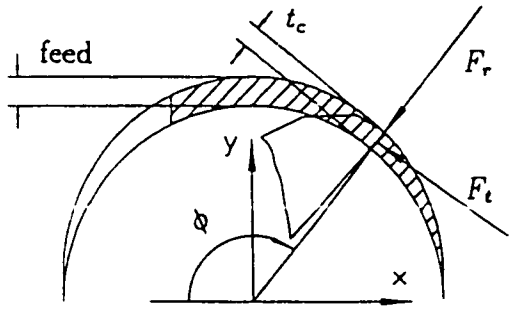


Figure 4.17: Differential cutting forces and undeformed chip thickness

4.3.2 Tool and structure dynamics

The vibratory system of the tool and machine structure has typically two dominant modes [61] : 1) The vibration mode of the end-mill which has higher natural frequency as well as higher flexibility; and 2) the vibration mode of the spindle with bearings acting as a spring. The typical natural frequency of the above modes are 1800 and 600 Hz, respectively (Figure 4.18).

A lumped mass-damper-spring model is proven to be adequate for simulating the tool structure dynamics [63]. Two degrees of freedom are considered in each of two non-rotating perpendicular directions (Figure 4.19). The dynamic characteristics are measured using modal testing performed on the machine tool (refer to Appendix B for the details).

In the following, the differential equations of motion in the local coordinate system are derived. The procedure is explained for the x direction, and the same argument can be followed for the other direction. The exciting force is the x component of the total cutting force applied on the tool (Figure 4.19). The displacement and force vectors as well as mass, damping, and stiffness matrices are denoted as follows:

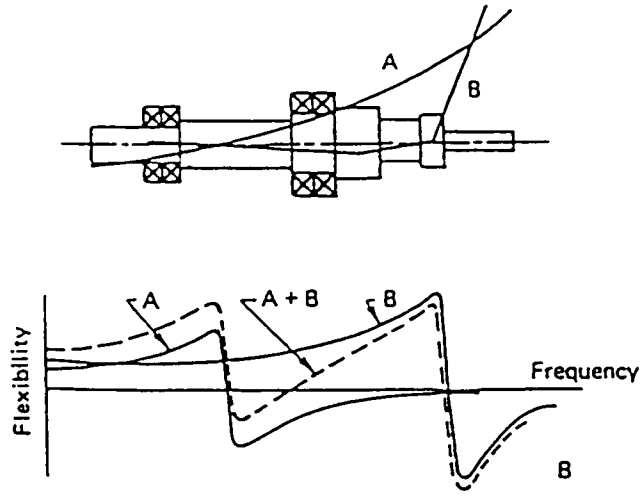


Figure 4.18: Spindle and tool mode shapes [61]

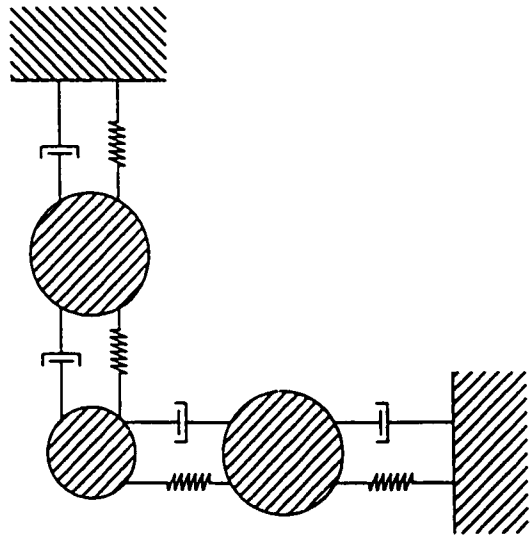


Figure 4.19: Lumped mass-damper-spring model

$$X = \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} \qquad F_x = \begin{Bmatrix} 0 \\ F_{x_2} \end{Bmatrix}$$

$$m_x = \begin{Bmatrix} m_{x_1} & 0 \\ 0 & m_{x_2} \end{Bmatrix}$$

$$c_x = \begin{Bmatrix} c_{x_1} + c_{x_2} & -c_{x_2} \\ -c_{x_2} & c_{x_1} + c_{x_2} \end{Bmatrix}$$

$$k_x = \begin{Bmatrix} k_{x_1} + k_{x_2} & -k_{x_2} \\ -k_{x_2} & k_{x_1} + k_{x_2} \end{Bmatrix} \tag{4.8}$$

The differential equation of motion can be express by

$$m_x\ddot{X} + c_x\dot{X} + k_xX = F_x \tag{4.9}$$

The solution of natural vibrations (a homogeneous system) leads to the determination of natural frequencies (ω_{n_1} and ω_{n_2}) and mode shapes ($(X_1/X_2)_{\omega_{n1}}$ and $(X_1/X_2)_{\omega_{n2}}$).

The modal matrix P_x is the transformation matrix between local coordinates X and the normal coordinates Q_x, given by

$$P_x = \begin{Bmatrix} \begin{Bmatrix} X_1/X_2 \\ 1 \end{Bmatrix}_{\omega_{n_1}} & \begin{Bmatrix} X_1/X_2 \\ 1 \end{Bmatrix}_{\omega_{n_2}} \end{Bmatrix} \tag{4.10}$$

$$Q_x = \begin{Bmatrix} q_{x_1} \\ q_{x_2} \end{Bmatrix} \tag{4.11}$$

$$X = P_xQ_x \tag{4.12}$$

With the above transformation, the vibrations of the tool in the local coordinate system can be obtained by adding the individual solutions in the modal coordinates, i.e.,

$$x_2 = q_{x_1} + q_{x_2} \tag{4.13}$$

In order to solve the system in the modal coordinate, the exciting force should be transformed into the modal coordinate:

$$R_x = P_x^T F_x \tag{4.14}$$

which results in

$$R_x = \begin{Bmatrix} F_{x_2} \\ F_{x_2} \end{Bmatrix} \tag{4.15}$$

Thus, the exciting force is acting equally on each mode in the modal coordinates. Finally, the equations of motion in the modal coordinates can be represented by

$$M_x \ddot{Q}_x + C_x \dot{Q}_x + K_x Q = R_x \tag{4.16}$$

The dynamic parameters of the above system of equations are measured by modal testing (Appendix B). Before numerically integrating the modal equations of motion in x and y directions, three important phenomena in the closed loop of the machine tool-cutting process must be taken into consideration.

4.3.3 Closed loop of machine tool-cutting process

One important mechanism affecting the machine tool-cutting process closed loop is the regeneration. The chip load on the cutting edge depends not only on the feed and the current deflection of the tool, but also on the surface which was cut by the preceding cutting edge [50]. The differential chip load considering the regeneration is given by

$$dA = (f_t + Z - Z_1)dz \tag{4.17}$$

Z and Z_1 are the current and the preceding edge normal-to-cut deflections, respectively.

However, as the vibration grows, more preceding cuts may be involved [60]. In the simulation three preceding cuts are taken into account. The displacement normal-to-cut denoted by Z_1, Z_2, and Z_3 are stored, and the lowest value at each instant is used in the chip load calculation.

$$\begin{aligned} h &= f_t \cos(\phi) \\ Z_1 &= Z(N-L) + h \\ Z_2 &= Z(N-2L) + 2h \\ Z_3 &= Z(N-3L) + 3h \\ Z_{min} &= min(Z_1, Z_2, Z_3) \\ t_c(N) &= Z_{min} - Z(N) \end{aligned} \tag{4.18}$$

It is obvious that, in the case of $t_c(N) < 0$, the cutting edge is out of the cut which

results in zero forces. This phenomenon is known as the basic non-linearity in the cutting process [60].

The next mechanism affecting the cutting process is the ploughing mechanism which occurring in the vicinity of the edge nose. The ploughing force is of interest not only in connection with the analysis of the balance of forces but also in explaining the amount of damping generated in the cutting process. Different approaches have been used to model the ploughing forces (for example, see [20]). In the current implementation the damping ratio of 0.05 for x_2 and y_2 are considered. The damping forces are applied when the tool tip is approaching the cut.

4.3.4 Identifying onset of chatter vibrations

An important issue for the detection of chatter vibrations is to develop a method which is sensitive to the transition between stable and unstable machining and insensitive to variations of cutting conditions as well as transient vibrations. For the in-process detection of the chatter, the ratio of acceleration signal variances is used. For the physical simulation, the onset of chatter is reported when the maximum cutting forces are increasing by ten percent.

The acceleration signal, collected by a Piezo-electric accelerometer mounted on the machine table along the y-axis, can be approximated by

$$a_y(t) = A\sin(\omega_p t + \theta) + n(t) \tag{4.19}$$

The first term is an approximation of the acceleration caused by the tooth-passing forces, while the second term represents a random noise. In the extreme case (unstable

machining), the random noise dominates the signal. These two terms can be easily recovered from the signal using two band-pass digital filters. The cut-off frequencies of the filters depend on the tooth-passing frequency and the dynamic characteristics of the tool, the spindle, and the structure.

The ratio of the first term variance to the noise variance $R = (\sigma_{sine})^2/(\sigma_{noise})^2$ can assist us to identify the onset of chatter. In Figure 4.20 [11], plots of the probability density for various ratios are shown. When $R >> 1$ the sinusoidal term is dominant which denoted stable cutting. On the other hand, if $R < 1$ the noise term is dominant which indicates the presence of unstable cutting or chatter vibrations.

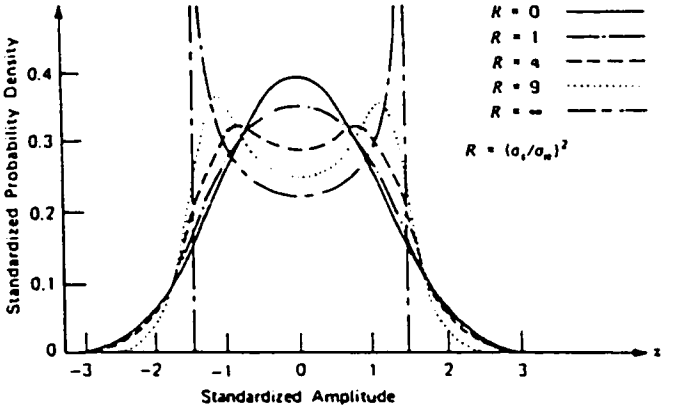


Figure 4.20: Standardized probability density function [11]

4.4 Experimentation and verification

In this section the simulations and experimentations which were conducted to calibrate the mechanistic force model and to anticipate the most stable direction of the machine tool for given cutting conditions are presented.

4.4.1 Experimental setup

Figure 4.21 illustrates a schematic representation of the equipment used. A vertical 3-axis machining center (YAM) was used for the roughing operations. The direction of x- , y-, and z-axis of motion are depicted the figure. To calibrate the force model a Kistler type 9255A three-force component table dynamometer was used. The dynamometer is relatively rigid with the flat frequency of 2000 Hz. In order to synchronize the sampling of instantaneous cutting forces (F_x, F_y, and F_z) with the angular position of the cutting edge, a shaft encoder was used as an external clock. Also, to detect the onset of chatter vibrations, a Kistler Piezo-electric accelerometer type 8702B25 was mounted on the table along the y-axis to collect data required for the detection of chatter.

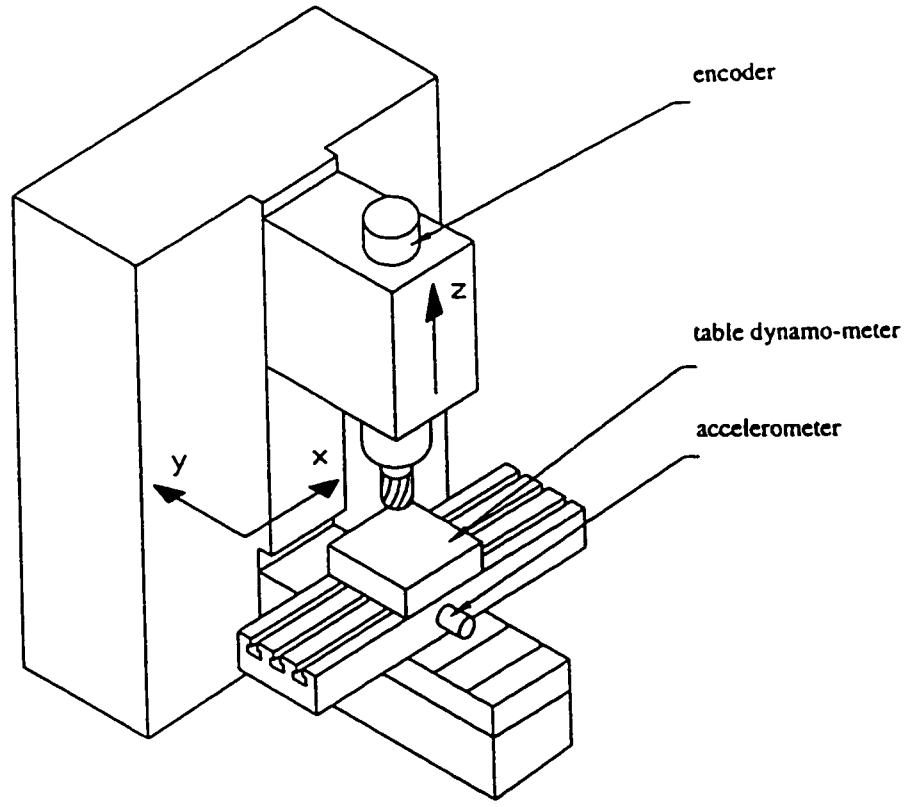


Figure 4.21: Experimental setup

4.4.2 Mechanistic force model calibration

The model calibration to determine the cutting coefficients K and c is performed for a one inch diameter four-flute flat-end mill. A new HSS tool is used to minimize the effects of the run-out and wear. The cutting conditions used for calibration are as follows: Down-milling is performed with $12.7mm$ WOC (half immersion) and $5mm$ DOC. The feed is set to 0.1 $mm/tooth$ throughout the experiments.

The cutting coefficients (K and c) estimated by curve fitting are given in Table 4.1.

K [N/mm^2]	1350
c	0.35

Table 4.1: Flat-end milling cutting coefficients

The agreement between the measured and predicted F_x is excellent (Figure 4.22). However, there is a small discrepancy between measured and predicted F_y which can be result of: 1) underestimating the edge forces (or ploughing forces) where the chip thickness is reaching to zero, and 2) the tool deflection perpendicular to the cut.

4.4.3 Onset of chatter vibrations

A series of cutting tests were conducted to experimentally find the the onset of chatter along different directions of the machine tool. The directions between 30 and 150 degrees with respect to x-axis are taken into consideration. The cutting conditions

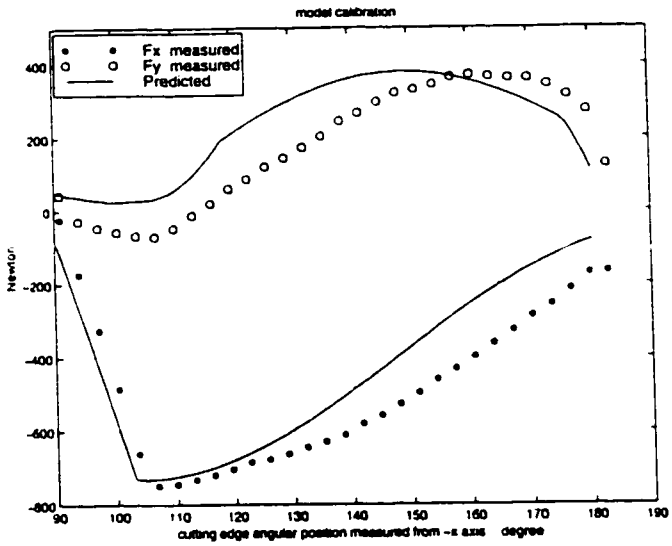


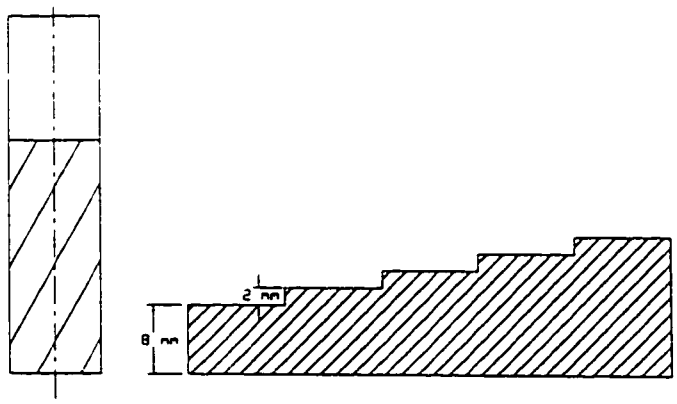
Figure 4.22: Flat-end milling model calibration

are as follows: Down-milling was performed with 3/4 immersion, $600rpm$ cutting speed, and $0.1mm/tooth$ feed. As depicted in Figure 4.23-a, the DOC for the first pass was set to $8mm$ and increased with the increment of $2mm$ until chatter vibrations are notified.

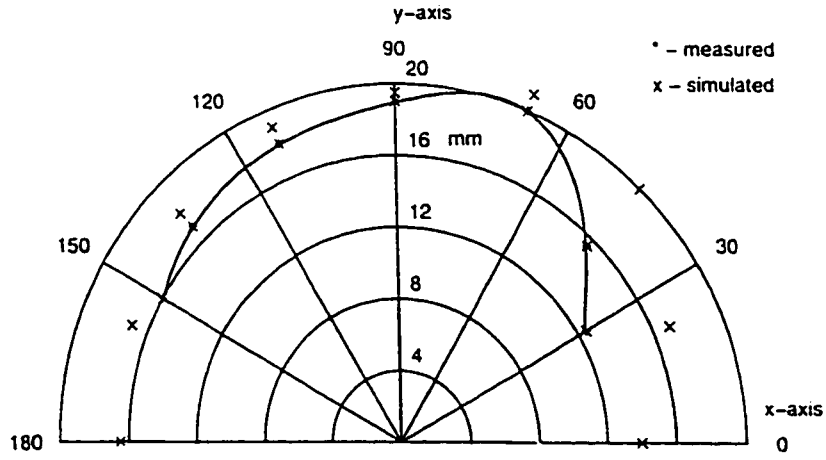
The measured onset of chatter is presented in Figure 4.23-b. The curve includes angles between 30 to 150 degrees. The most stable direction is found to be 70 degrees. Also, in the same figure the onset of chatter predict by simulation is presented. The estimation along the y-axis is closed to the measurement; however, along the 45-degree line the onset is overestimated.

4.5 Feed scheduling

The pre-process modification of the feed subjected to a set of constraints has been considered as one of the key elements in the cutting process optimization [10][57]. For



a- variation of depth of cut



b- onset of chatter at different orientations

Figure 4.23: Onset of Chatter vibrations

the flat-end milling of the contour path (see Section 4.1), the dominant constraints are: 1) the tool deflections over the whole length of the contour path and 2) the chatter vibrations where heavy chip load is encountered.

The flat-end mill deflections, tangent- and normal-to-path, are estimated as follows: First, the static stiffness of the structure along x and y directions at the tool tip (k_x and k_y) are computed, see Appendix B for details. Then, based on the predicted maximum cutting forces maximum tool deflections (δ_x and δ_y) are computed. Finally, the x and y deflections are decomposed into the tangent- and normal-to-path directions. The feed along the contour path is adjusted based on the tangent-to-path deflection constraint in order to avoid surface gouging. The maximum allowed deflection tangent-to-path depends on the tool path curvature. For example, in the corner areas where the curvature is large, a tighter defection is allowed.

The chatter vibration constraint has been investigated and applied to face milling operations [10]. The same approach is used here for heavy chip load areas of the contour path. First, a near to optimum feed is computed using the above static model considering the deflection constraint. Then, the feed subjected to the chatter vibration constraint is computed using the dynamic regenerative model.

4.6 Summary

In this chapter, the pocketing simulation methodology has been developed for the die cavity roughing operation and implemented in the form of a model-based pocketing simulation system. The system consists of geometric and physical simulation modules. The geometric module accurately updates the B-rep model of the stock as the

cutter removes the material and automatically extracts the immersion geometry from the B-rep updated model. The optimum tool path pattern for die cavity pocketing operations is proposed. The physical simulation module, integrated with the geometric module, is a regenerative mechanistic force model which simulates the closed loop of the machine tool-cutting process. It can accurately predict the instantaneous cutting forces as well as maximum tool tip deflections. The physical module can also anticipate the onset of chatter vibrations. The applications of the developed system in the feed scheduling of the contour path and anticipating the most stable direction of the machining unit are discussed.

Chapter 5

Two- and three-axis semi-finishing

The model-based simulation methodology for the die cavity semi-finishing operation is proposed in this chapter. In the first section, the two-axis tool paths with ball-nose cutters are computed using the updated B-rep model of the rough-cut part. The three-axis tool paths are also introduced in order to control the scallop height distribution and to impose the dead zone constraint. The next section deals with the geometric simulation issues of the semi-finishing. A three-axis swept volume generation technique is developed which is central to the part updating process. Also, the undeformed chip geometry is precisely extracted from the updated B-rep model. The physical simulation module developed in Section 5.3 is a new three-axis mechanistic force model which predicts the instantaneous cutting forces. The applications of the developed system in predicting the cutting forces as well as feed scheduling are discussed in the following two sections. The last section summarizes the chapter and accomplishments.

5.1 Tool path based on updated B-rep model

In order to improve the MRR of the semi-finishing operation, the 2-axis tool path generation techniques based on the B-rep model of the rough-cut part and the design surface have been implemented (e.g., [56]). The important constraint in this regard is to keep the ball-nose tip (the dead point) out of the cut to avoid edge chipping as well as poor surface quality.

5.1.1 Two-axis tool path

The rough-cut B-rep model and the design surface are the geometric information required for computing semi-finishing two-axis tool paths. First, the offset of the design surface is computed. Then, the offset face is intersected with the shoulder offset face (Figure 5.1), which represents the cutter center points. The intersection curve is a two-axis tool path which assures that the ball-nose tip (the dead point) is not engaged.

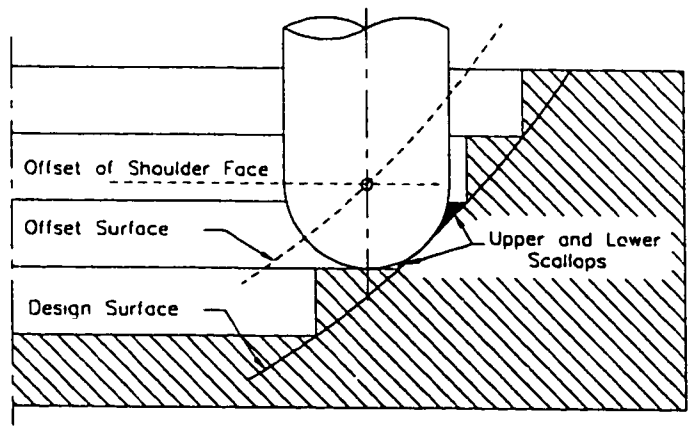


Figure 5.1: Two-axis tool path and left scallops

Removing a shoulder with a ball-end mill leaves two scallops on the surface as

depicted in Figure 5.1. The scallop height, the normal distance between the scallop peak and the design surface, is computed as follows. First, a semi-circle with ball-nose radius is oriented perpendicular to the tool path. Then, the semi-circle is intersected with the vertical and horizontal faces of the shoulder to obtain the scallop peak points. The peak points are projected to the design surface in order to compute the normal distance between the scallop peak and the design surface. Figure 5.2 shows the scallop height variation along the two-axis tool path.

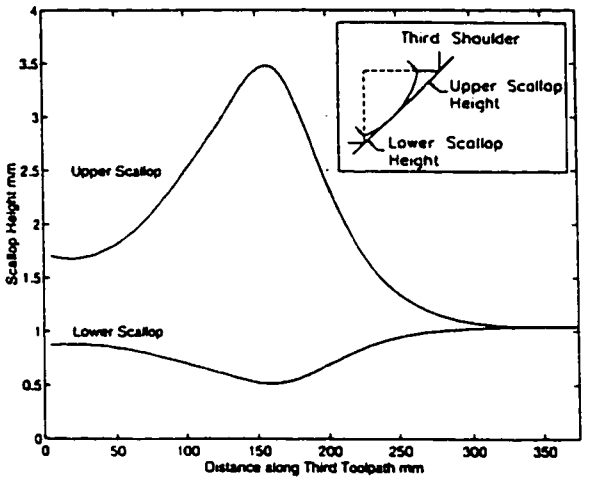


Figure 5.2: Scallop height distribution for two-axis tool path

The limitation imposed by the dead point constraint as well as by two-axis motions results in a poor scallop height distribution on the part surface. In general, the part is heat-treated and reaches a level of higher hardness before the finishing operation. For example, the hardness of tool steel H13 is roughly 35RC before heat-treatment but it can reach more than 65RC afterwards. Large scallops together with high material hardness can cause serious problems for the finishing operation such as excessive tool deflections, chipping of the cutting edge, and a poor surface quality. The idea of using a three-axis tool path in order to control the scallop height is investigated in the next

section.

5.1.2 Scallop height control

The three-axis CNC controllers are widely replacing $2\frac{1}{2}$-axis controllers on the milling machines. Therefore, in most cases without any change in the dynamic characteristics of the machine tool structure, the limited two-axis tool paths can be replaced by more advantageous three- axis tool paths. The main advantages of using a three-axis tool path are as follows:

1. Equal scallop heights within a prescribed value can be obtained.

2. The dead zone constraint, which is the segment on the ball-nose tip within 10 degrees from the vertical axis(Figure 5.3), can be considered.

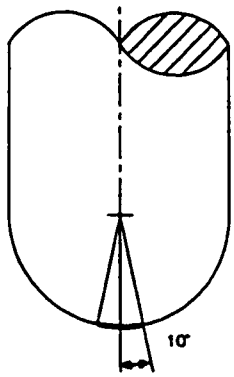
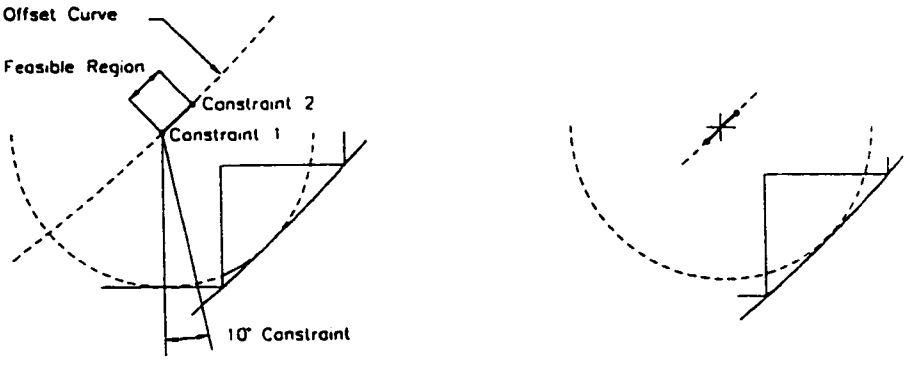


Figure 5.3: Dead zone constraint

Three-axis tool path calculation

Having computed the two-axis tool path as mentioned in Section 5.1.1, the three-axis tool path can be computed as follows: First, a plane normal to the two-axis tool path

is constructed. Then, the normal plane is intersected with the design offset surface. The intersection curve (the offset curve) is the locus of ball-nose center point. In the next step, the cutter center constraints are computed. As illustrated in Figure 5.4-a, Constraint 1 is the tool tip constraint. At any distance below this point, the tool tip cuts the upper face of the next shoulder. The second constraint is the dead zone constraint (Constraint 2). At any distance above this point, the 10 degree zone is engaged. Thus, the feasible region of the offset curve is bracketed by Constraint 1 and 2. The Bisection method is used to find the optimum position of the cutter center point where the two scallop heights are equal (Figure 5.4-b). The method starts with two values of the offset curve parameter at Constraint 1 and 2. Then, it successively divides the interval in half and replaces one endpoint parameter with the midpoint parameter so that, again, the solution is bracketed. The solution can be either a point with two equal scallop heights or a point on the boundary of the feasible region. Lastly, the computed cutter center points are interpolated by a cubic



a- feasible region b- optimum cutter center position

Figure 5.4: Scallop height control

NURBS curve to represent the three-axis tool path.

$$\mathbf{T}(v) = \frac{\sum_{i=0}^{m'} w_i \mathbf{Q}_i \mathbf{N}_{i,3}(v)}{\sum_{i=0}^{m'} w_i \mathbf{N}_{i,3}(v)} \tag{5.1}$$

where $N_{i,3}$, ω_i are cubic B-spline basis function and weights, respectively and Q_i are the control points computed by interpolation. In the actual machining, the NURBS tool path is approximated by a series of linear segments assuming a linear interpolation by the CNC controller.

Scallop height distribution

The scallop height distribution for the third shoulder is computed by the method discussed in 5.1.1 and depicted in Figure 5.5. The objective of decreasing the scallop height to less than a prescribed value (for this example 1.6 mm) has been achieved for portions 1-2 and 3-4 of the tool path. However, the scallop height in the corner area (portion 2-3) exceeds the desired value since the cutter center point is on the boundary of feasible region. For this portion, an extra tool path is proposed.

The feasible region for the extra tool path on the offset curve is wider. The constraint is the dead zone constraint and the objective is again the equal scallop height. The solution is found in a manner similar to that used to generate the first tool path. Figure 5.6 illustrates the scallop height distribution after the extra tool path for the corner area has been performed. In the corner area the upper scallop is removed and two equal small scallops are left on the surface.

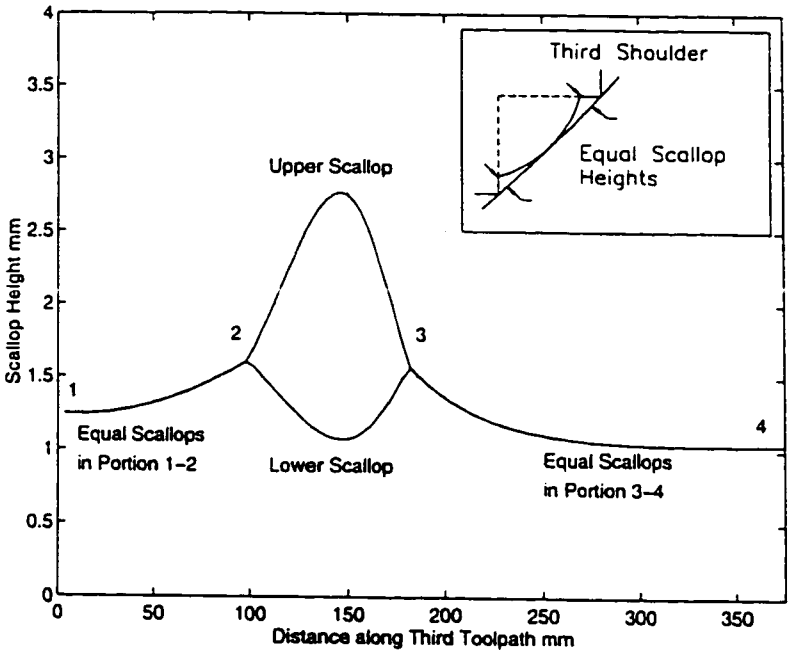


Figure 5.5: Scallop height distribution for three-axis tool path

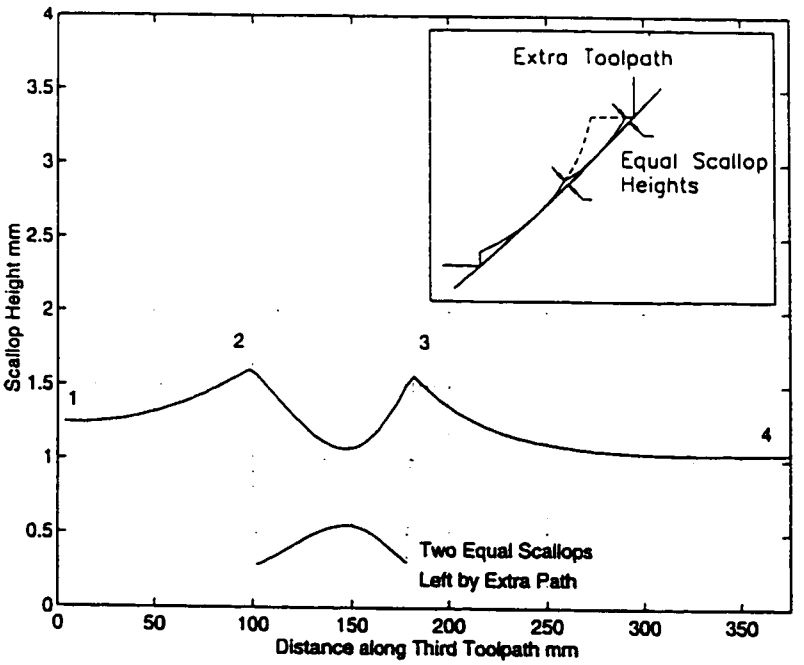


Figure 5.6: Scallop height distribution after performing extra tool path

5.2 Geometric simulation of ball-end milling

The geometric modeling issues of the semi-finishing operation are discussed in this section. A new three-axis swept volume generation method is developed in Section 5.2.1 which can be applied to any form cutter tool. The part updating and tool path verification for one quarter of the die with sculptured surfaces are implemented in Section 5.2.2. The last section deals with the instantaneous chip geometry extraction from the B-rep model of the updated part.

5.2.1 Three-axis swept volume generation

The kinematics of the milling operation consists of a rotational motion of cutting edges and translational motions of the workpiece. In addition, depending on the application, cutting edge run-out and deflections can be considered. This system is replaced by an equivalent system in which the workpiece is stationary and the cutting edges have simultaneous rotational and translational motions. Thus, the true path of points on the cutting edge for 2-axis milling is trochoidal, as demonstrated by Martellotti [40]. Since the material is removed by a trochoidal path of consecutive cutting edges, cusps (or feed marks) with the spacing of feed per tooth are generated on the machined surface (Figure 4.5).

To precisely simulate the material removal of milling operations, the volume removed must be constructed based on the true path of each point on the cutting edge. However, for the non-functional surfaces, the cusp height is either one or two order of magnitude smaller than dimensional tolerances [27]. Therefore, the true swept volume can be simplified as the volume generated by a translational sweeping motion of

a solid of revolution, which was proposed by Wang [70] and Menon [42]. For the case of ball-end milling the solid of revolution (i.e., the cutter), consists of a cylinder and a semi-sphere representing the envelope of rotating cutting edges on the cylindrical and spherical parts of the ball-end mill.

In general, ten faces are required for the construction of a valid 3-axis cutter swept volume, as shown in Figure 5.7. The following argument is followed with the assumption of a rigid tool. However, knowing the tool axis deflection curve, the same argument can be followed to generate the swept volume for a flexible tool. The boundary of swept volume consists of two categories of faces. The first category which consists of analytical faces (FACEs 5 to 10) represents portions of the cutter at initial and final positions. These planar, cylindrical, or spherical faces are supported in the geometric kernel (ACIS). The second category, FACEs 1 to 4, is tool path dependent. The current research proposes to use advanced sweeping/skinning techniques to construct precise NURBS representation of these faces which results in a higher level of precision.

In the following, the detailed procedure of the construction of FACE 1 which represents the swept surface by the ball-nose will be discussed.

A general form of a swept surface is given by 2.7 and repeated here

$$\mathbf{s}(u, v) = \mathbf{T}(v) + \mathbf{A}(v)\mathbf{C}(u) \tag{5.2}$$

where $\mathbf{C}(u)$, $\mathbf{T}(v)$, and $\mathbf{A}(v)$ denote the section curve, path curve, and general transformation matrix, respectively. Thus, iso-parametric curves on the swept surface $\mathbf{s}(u, v)$ are instances of $\mathbf{C}(u)$ transformed by $\mathbf{A}(v)$ and translated by $\mathbf{T}(v)$. As $\mathbf{A}(v)$

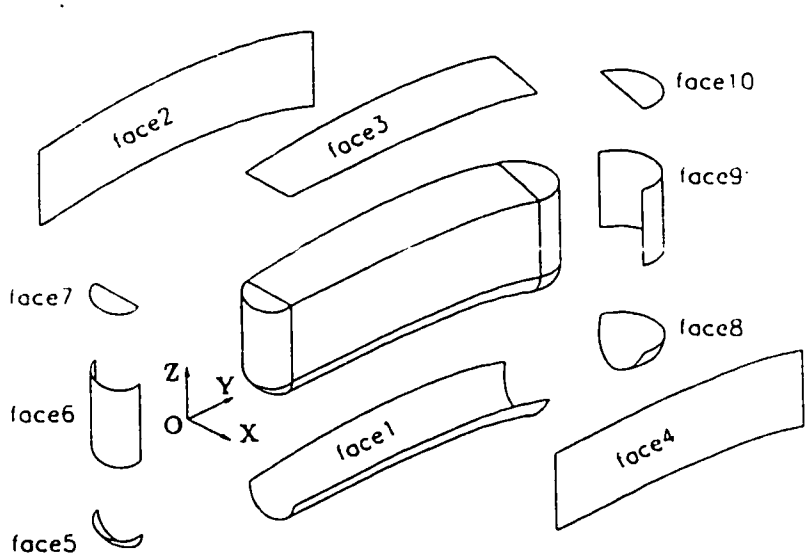


Figure 5.7: Three-axis swept volume and its boundary faces

is generally not representable in a NURBS form, a NURBS approximation to $s(u, v)$ must be constructed. This research adopts advanced skinning techniques in which the accuracy of the resulting surface is improved by imposing path derivative constraints at each data point.

For FACE 1 the section curve, $C(u)$, is a semi-circle. In general, conic sections can be exactly represented in quadratic rational B-splines[46]. However, there is no obvious technique to impose path (or v-direction) derivative constraints for rational B-splines [47]. Due to this limitation, quadratic non-rational B-splines are chosen to interpolate the semi-circle. Thus, the interpolated section curve can be expressed by

$$\mathbf{C}(u) = \sum_{i=0}^{n} \mathbf{P}_i \mathbf{N}_{i,2}(u) \tag{5.3}$$

The B-spline representation of the section curve also allows us to implement the method for any form cutter tool.

It is assumed that the tool path (path curve) is given and represented in cubic NURBS form :

$$\mathbf{T}(v) = \frac{\sum_{i=0}^{m'} w_i \mathbf{Q}_i \mathbf{N}_{i,3}(v)}{\sum_{i=0}^{m'} w_i \mathbf{N}_{i,3}(v)} \tag{5.4}$$

$K+1$ instances of $\mathbf{C}(u)$ are transformed and positioned along $\mathbf{T}(v)$ at parameter values of $\{v_k\}$, $k = 0, \ldots, K$. The $\{v_k\}$ spacing affects the skinning tolerance. For the current implementation equal spacing is considered.

To construct $\mathbf{A}(v)$, a local Cartesian coordinate system $\{\mathbf{o}(v_k), \mathbf{x}(v_k), \mathbf{y}(v_k), \mathbf{z}(v_k)\}$ in terms of the global coordinate system, $\{\mathbf{O}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}\}$ is introduced. The origin of the local coordinate system is placed at $\mathbf{T}(v_k)$ and three orthonormal unit vectors are defined as follows. $\mathbf{y}(v_k)$ is the unit vector tangent to the path in the feed direction. $\mathbf{x}(v_k)$ is a horizontal unit vector normal to $\mathbf{y}(v_k)$. Lastly, $\mathbf{z}(v_k)$ is a unit vector which makes a right-hand coordinate system with x and y :

$$\begin{aligned} \mathbf{o}(v_k) &= \mathbf{T}(v_k) \\ \mathbf{y}(v_k) &= \frac{\mathbf{T}'(v_k)}{|\mathbf{T}'(v_k)|} \\ \mathbf{x}(v_k) &= \frac{\hat{\mathbf{k}} \times \mathbf{y}(v_k)}{\left|\hat{\mathbf{k}} \times \mathbf{y}(v_k)\right|} \\ \mathbf{z}(v_k) &= \mathbf{x}(v_k) \times \mathbf{y}(v_k) \end{aligned} \tag{5.5}$$

For further details refer to Siltanen and Woodward [49]. Having constructed the local coordinate system (see also Figure 5.8), we can compute the general transformation matrix, $\mathbf{A}(v_k)$. Rows of $\mathbf{A}(v_k)$ are the unit vectors of the local coordinate system :

$$\mathbf{A}(v_k) = \{\mathbf{x}(v_k), \mathbf{y}(v_k), \mathbf{z}(v_k)\}^{\mathrm{T}} \tag{5.6}$$

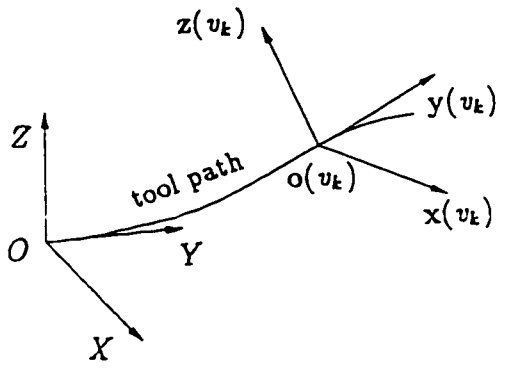


Figure 5.8: Global and local coordinate system

In the next step, control points of the section curve, $\mathbf{P}_i$, are transformed by $\mathbf{A}(v_k)$ to orient the section curve perpendicular to the tool path and then translated by $\mathbf{T}(v_k)$ to place the center of section curve at $\mathbf{o}(v_k)$:

$$\mathbf{P}_{i,k} = \mathbf{T}(v_k) + \mathbf{A}(v_k)\mathbf{P}_i \tag{5.7}$$

where $\mathbf{P}_{i,k}$, $i = 0, \ldots, n$, are control points of the kth semi-circle instance. Then, for each i and k a derivative vector, $\mathbf{D}_{i,k}$, is computed based on the tangent to the tool path at v_k, i.e.

$$\mathbf{D}_k = \mathbf{T}'(v_k) \tag{5.8}$$

All $\mathbf{D}_{i,k}$ are parallel to $\mathbf{D}_k$ and their magnitudes are adjusted according to the following relations to reflect relative distance

$$\begin{aligned} d_k &= |\mathbf{P}_k - \mathbf{P}_{k-1}| \\ \mathbf{D}_{i,0} &= \frac{|\mathbf{P}_{i,1} - \mathbf{P}_{i,0}|}{d_1}\mathbf{D}_0 \end{aligned}$$

$$\begin{aligned}
\mathbf{D}_{i,K} &= \frac{|\mathbf{P}_{i,K} - \mathbf{P}_{i,K-1}|}{d_K}\mathbf{D}_K \\
\mathbf{D}_{i,k} &= \frac{|\mathbf{P}_{i,k+1} - \mathbf{P}_{i,k}| + |\mathbf{P}_{i,k} - \mathbf{P}_{i,k-1}|}{d_{k+1} + d_k}\mathbf{D}_k \qquad (5.9)\\
&\quad {\scriptstyle i=0,\ldots,n} \\
&\quad {\scriptstyle k=0,\ldots,K}
\end{aligned}$$

Imposing the derivative constraints based on the above equations ensures that all partial derivatives with respect to v on the kth iso-parametric curve are parallel to $\mathbf{D}_k$ [47]. For the v-directional interpolation, in addition to the degree ($p = 3$), the parameter values of interpolating points $\{\bar{v}_k\}$ and the knot vector $\mathbf{V} = \{v_i\}$ are required. The $\bar{v}_k$ are computed by the averaging method

$$\begin{aligned}
\bar{v}_k &= \bar{v}_{k-1} + \frac{1}{n+1}\sum_{i=0}^{n}\frac{|\mathbf{P}_{i,k} - \mathbf{P}_{i,k-1}|}{d_i} \qquad (5.10)\\
&\quad {\scriptstyle k=0,\ldots,K\text{-}1} \\
&\quad \bar{v}_0 = 0 \quad \bar{v}_K = 1 \\
d_i &= \sum_{k=1}^{K}|\mathbf{P}_{i,k} - \mathbf{P}_{i,k-1}|
\end{aligned}$$

A satisfactory choice of the knot vector considering the distribution of $\{\bar{v}_k\}$ is

$$\begin{aligned}
\mathbf{V} = \{0,0,0,0,&\frac{\bar{v}_1}{2}, \frac{2\bar{v}_1 + \bar{v}_2}{3}, \frac{\bar{v}_1 + 2\bar{v}_2}{3}, \ldots, \\
&\frac{\bar{v}_{K-2} + 2\bar{v}_{K-1}}{3}, \frac{\bar{v}_{K-1} + 1}{2}, 1,1,1,1\} \qquad (5.11)
\end{aligned}$$

The last step in skinning FACE 1 is to solve the following system of linear equations for each i. There are $2(K+1)$ known data items (i.e., $\mathbf{P}_{i,k}$ and $\mathbf{D}_{i,k}$) and $2(K+1)$

unknown control points $\mathbf{Q}_{i,j}$,$j = 0, \ldots, m$, $(m = 2K + 1)$

$$\begin{aligned} \mathbf{P}_{i,k} &= \sum_{j=0}^{m} \mathbf{Q}_{i,j} \mathbf{N}_{j,p}(\bar{v}_k) \\ \mathbf{D}_{i,k} &= \sum_{j=0}^{m} \mathbf{Q}_{i,j} \mathbf{N}'_{j,p}(\bar{v}_k) \end{aligned} \qquad (5.12)$$

$k = 0, \ldots, K$

$i = 0, \ldots, n$

Finally, the swept/skinned surface representing FACE 1 can be expressed by

$$\hat{\mathbf{s}}(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} \mathbf{Q}_{i,j} \, \mathbf{N}_{i,2}(u) \, \mathbf{N}_{i,3}(v) \qquad (5.13)$$

The same argument can be followed for sweeping/skinning FACEs 2 to 4 except the section curves representing the cylindrical part of the cutter are linear.

After generating all the boundary FACEs, an ACIS "stitching" operation is used to generate a topologically valid swept volume. The validity of the swept volume is verified by checking its topology before performing a Boolean subtraction between the B-rep model of the part and the cutter swept volume.

5.2.2 Tool path verification

One quarter of the die updated for the roughing operation (Figure 4.13), is used in this section to illustrate the steps of the part updating for the 3-axis ball-end milling operation. The right angle shoulders remained on the part surface are precisely represented by the B-rep model.

In the part updating process, the swept volume is constructed first. The B-spline

tool path used for the swept volume represents a few blocks of the NC code. The number of blocks depends on the part geometry. The Boolean subtraction between part and swept volume is then performed which consecutively updates the part (Figure 5.9). Thus, it is possible to observe the precise simulation of the machining operation step-by-step. And, in case of any error, corrective actions can be taken before actual machining.

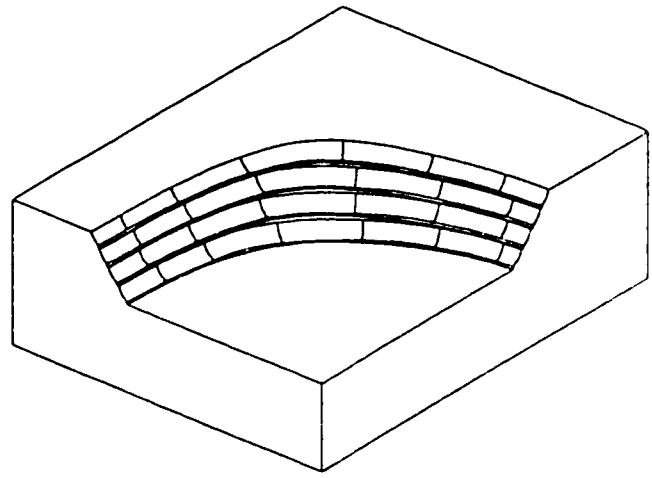


Figure 5.9: Semi-finished step-by-step updated part

During part updating, consecutive Boolean subtractions increase the number of FACEs in the updated part data structure as illustrated in Figure 5.9. As a result, the query time required for subsequent operations increases. For example, it is required to find the intersecting pair of FACEs before performing the face-face intersection of the Boolean operation. A fast algorithm for reporting all intersecting pairs first constructs the iso-box of each face, including the FACEs of the updated part and the query face, and then reports the intersecting pairs. Based on the sweep paradigm, the query time for reporting all intersection is $O(n\,log^2(n) + J)$ [30], where n is the

number of FACEs in the updated part and J is the number of intersections. In the developed simulator, the NURBS surfaces generated during consecutive updating are unified and represented by one NURBS surface for each tool path (Figure 5.10). Thus, the size of the B-rep data structure decreases which results in faster query time.

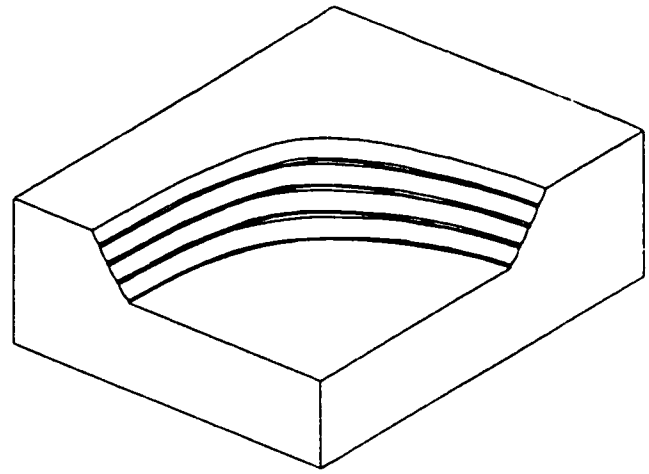


Figure 5.10: Semi-finished updated part with unified FACEs

The semi-finished B-rep model of the part includes valuable information for tool path planning of the following operation, that is, finishing. The scallop geometry left by the ball-nose cutter can be precisely represented by B-rep NURBS enhanced solid modeling techniques. Thus, the scallop height defined as the normal distance between the scallop peak and the design surface can be extracted from the B-rep model (see Section 5.1.2). The B-rep model, therefore, can assist the finishing tool path planning in order to achieve higher metal removal rates (MRR), smaller scallop heights, and longer tool life.

5.2.3 Instantaneous undeformed chip geometry

The required geometric information for the physical simulation of the machining process includes: 1) tool geometry (such as the cutting edge design) and 2) chip geometry, i.e., the in-cut segment of the cutting edge and undeformed radial chip thickness. The procedure for determining the above geometric information for the semi-finishing operation follows here.

Tool geometry

Tool geometric information can either be provided by the manufacturer or by direct measurement of points on the cutting edge and rake face and fitting a curve or surface (i.e., B-spline curve or surface) to the measured points. In this research a constant lead HSS ball-end milling cutter (Figure 5.11), which is widely used for semi-finishing operations, has been utilized. The cutter geometry is mentioned in Table 5.1. This constant lead geometry ($\frac{\text{cutter diameter} \times \pi}{\tan(\phi_0)} = Lead$) is not supported by the geometric modeler. Thus, it is replaced with a cubic piecewise NURBS curve. Ten points on the edge together with initial conditions (i.e., slopes of the edge at start and end points) are used to interpolate the cutting edge.

Chip geometry

In 3-axis machining of sculptured surfaces, the chip geometry is constantly varying along the tool path. In order to obtain accurate and reliable chip geometry, it is essential to construct a B-rep model of the workpiece and update it as the tool removes material. In addition to the updated workpiece, the chip geometry depends on the cutting edge design, the tool path, and the feed.

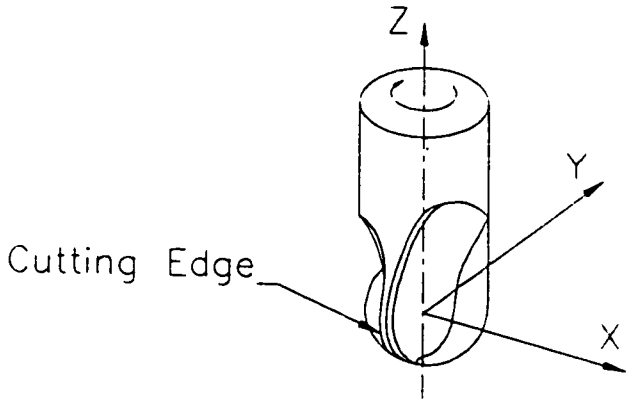


Figure 5.11: Ball-end mill with constant lead design

Ball-nose diameter	[mm]	25.4
Tool length	[mm]	100
Helix angle at start (measured)	[degrees]	22.5
Rake angle at start	[degrees]	6
Tolerance of ball radius	[micron]	5

Table 5.1: Ball-end mill geometry

The instantaneous in-cut segment which is the portion of the cutting edge engaged during machining is computed in two steps. First, the boundary of the contact face between the ball-end mill and the updated part is constructed. In two- or three- axis ball-end milling this boundary consists of two B-spline edges and one circular edge (Figure 5.12). The B-splines are intersections of the cutter with faces of the updated part. The circular edge is the boundary where the cutting edge either leaves or enters the cut. This edge is constructed using the B-spline endpoints, the cutter radius and the slope of tool path. The second step is to determine where the interpolated cutting

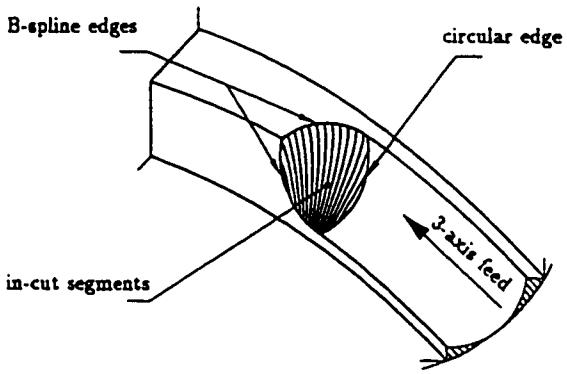


Figure 5.12: Instantaneous in-cut segment extraction

edge intersect with the boundary of the contact face. The edge-edge intersection function finds the closest point on the edges with a prescribed tolerance of 0.01 mm. The segments on the cutting edge are then classified into in-cut and out-of-cut segments. The in-cut segments are shown in Figure 5.12. The endpoints of each in-cut segment represent the integration limits required for the physical simulation of the machining process.

The simplified relationship (Equation 4.4) proposed by Martellotti [40] is not valid

for two- and three-axis ball-end milling any more. For the case of three-axis ball-end milling [25][38] considered the effect of the vertical component of feed on the chip thickness using the following simplified relationship

$$t_c(\theta, z) \approx R_2(z) - R_1(z) + f_h \sin\theta \tag{5.14}$$

f_h is the horizontal component of the feed along the tool path. R_1 and R_2 are the ball-nose radii at two successive positions in the same z value as shown in Figure 5.13. However, as θ decreases, the amount of error in the above approximation increases.

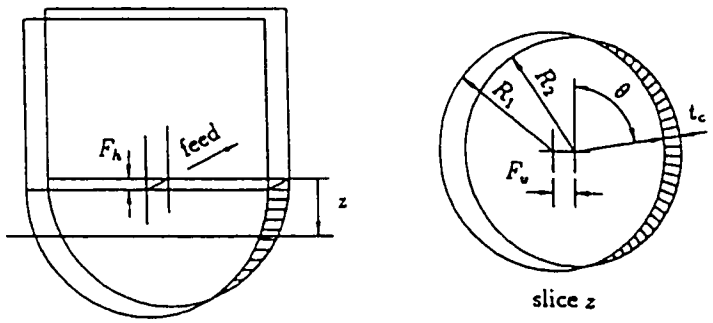


Figure 5.13: Variation of chip thickness for three-axis feed

Figure 5.14-a and -b depict the approximation errors for θ equal to 90, and 45 degrees, respectively. The accurate representation is obtained by skinning a B-spline surface to instances of the cutting edge while rotating with the cutting speed and translating with the feed (refer to Section6.1 for further details). The accurate chip thickness can then be obtained using two successive surfaces representing the accurate cutting edge motion. However, the extraction of instantaneous chip thickness from the accurate representation is computationally inefficient. Thus, a modified equation which can

accurately calculate the chip thickness at any angular position is used :

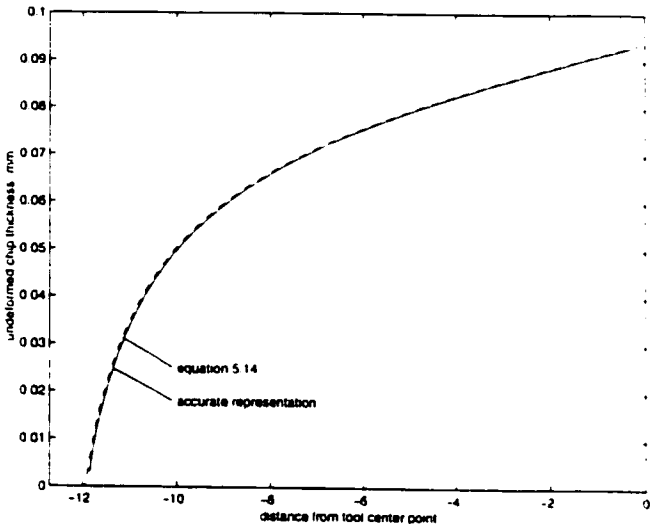
$$t_c(\theta, z) = R_2(z) + f_h \sin\theta - [R_1^2(z) - f_h^2 \cos^2(\theta)]^{\frac{1}{2}} \tag{5.15}$$

Figure 5.14-b also shows that with respect to the value obtained by accurate representation the amount of error in the above approximation is negligible at θ equal to 45 degrees.

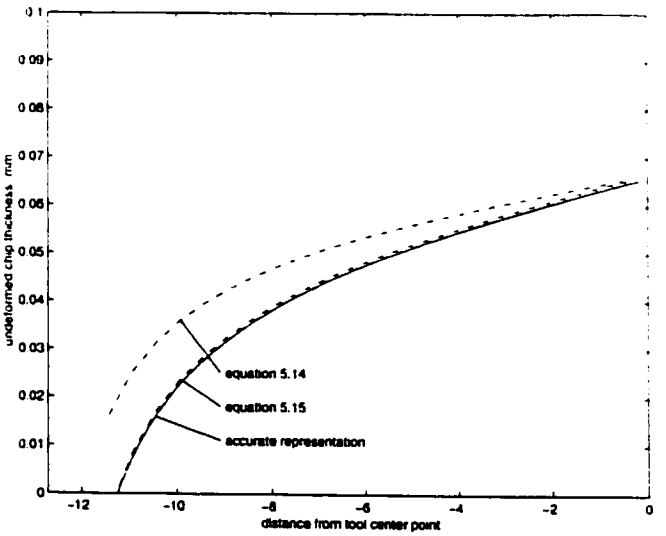
5.3 Physical simulation of ball-end milling

Cylindrical flat-end mill cutters have constant helix, rake angle and uniform cutting speed along their cutting edge. Unlike flat-end mills, ball-end mill cutters have variable effective cutting radius, helix and radial rake angles along the cutting edge. A typical ball-end mill cutter with constant lead cutting edge is shown in Figure 5.11. Since the effective cutting radius is decreasing in axial direction, the cutting speed is reducing along the edge, starting with the same value as the cylindrical portion and reaching to zero at the tool tip. The chip flow direction is also affected by this varying tool geometry and variable cutting conditions. Thus, the determination of instantaneous cutting forces in ball-end milling is rather difficult as opposed to the prediction of forces in flat-end milling. The research work reported in modeling the mechanics of ball-end milling are reviewed in Section 2.3.

In this part of the current work a new three-component mechanistic force model is developed to calculate instantaneous ball-end milling cutting forces. This force model not only takes into account the geometry of the cutting edge (rake and helix



a- chip thickness at 90 degrees



b- chip thickness at 45 degrees

Figure 5.14: Chip thickness: comparison of simplified and accurate representation

angles) but also considers the variations of the chip flow angle and cutting coefficients in the axial direction. Due to the large variation of cutting edge geometry, it is most important to consider the local mechanics of cut by introducing the chip flow angle and cutting coefficient as functions of the z coordinate. The chip flow angle and cutting coefficients are identified from half-immersion calibration tests, in which the whole length of the cutting edge is engaged. The model utilizes the geometric information extracted by the geometric simulation module (i.e., in-cut segment and undeformed chip thickness) to predict the instantaneous cutting forces for practical two- or three-axis ball-end milling operations. The model development and the detailed formulation are presented in the next section.

5.3.1 Mechanistic force model development

The cutting forces are considered as the summation of normal and friction forces acting on the rake face and the cutting edge. The unit vectors representing the directions of the differential forces are shown in Figure 5.15. It is assumed that the

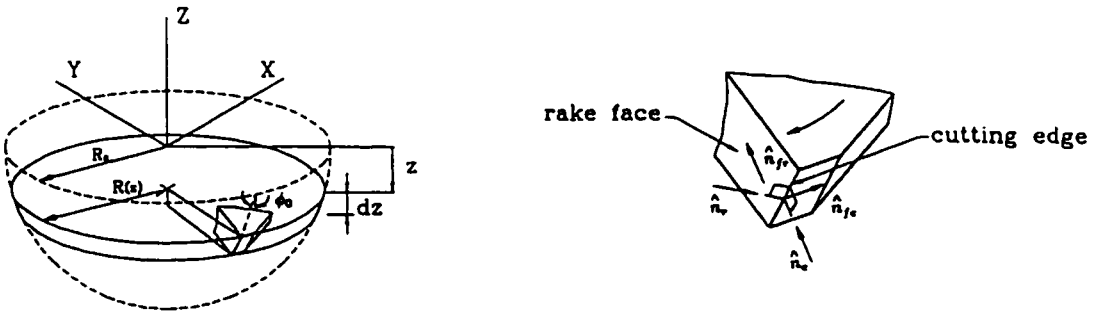
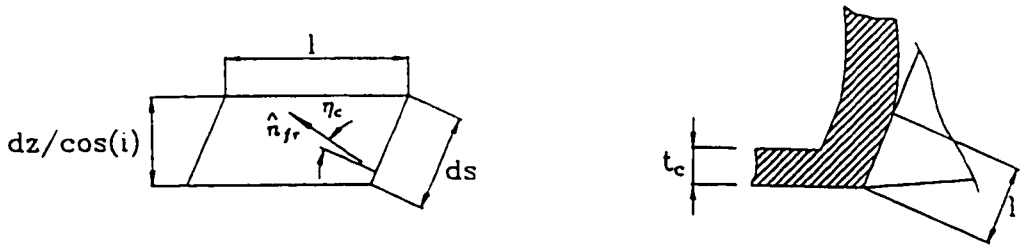


Figure 5.15: Unit vectors on rake face and cutting edge

friction force on the rake face, acts in the chip flow direction as depicted in Figure 5.16-a (the co-linearity condition). The chip flow angle η_c depends on the tool geometry, workpiece material and cutting conditions. The variation of the chip flow angle in z direction is assumed to have a linear characteristic with axial direction as follows:

$$\eta_c = c_1 + c_2 z \tag{5.16}$$



a- true view of rake face b- chip contact length

Figure 5.16: Differential chip contact area

The differential forces acting on the rake face $d\vec{F}_{nr}$ and $d\vec{F}_{fr}$ are assumed to be proportional to the differential chip contact area dA_r. The chip contact area is a function of the cutting conditions, workpiece material, tool geometry, and coolant. Okushima and Hitomi[44] suggested that the chip contact length is equal to twice the uncut chip thickness when cutting at low speeds, roughly less than 30 m/min. For higher cutting speed and ductile material, Bhattacharyya[13] investigated the seizure contact length based on the work done by Lee and Shaffer [36] on chip formation. For

cutting speed in the range of 30 to 60 m/min, the ratio of seizure contact length to chip thickness is roughly equal to three. In the current study, the cutting speed along the edge of the ball-nose varies from 0 to about 32 m/min. Thus, it is assumed that the chip contact length is equal to twice of the undeformed chip thickness, t_c (Figure 5.16-b). Therefore, the differential chip contact area is given by

$$dA_r(\theta, z) = 2t_c(\theta, z)\frac{dz}{\cos(i(z))} \tag{5.17}$$

where the helix angle $i(z)$ is expressed as follows

$$i(z) = \arctan(\frac{R(z)\tan(\phi_0)}{R_0}) \tag{5.18}$$

$R_0, R(z)$ and ϕ_0 are tool radius, slice radius and helix angle, respectively at the starting point.

Therefore, the differential normal and frictional forces on the rake face can be calculated by

$$\begin{aligned} \left|d\vec{F}_{nr}(\theta, z)\right| &= K_{nr}(z)dA_r(\theta, z) \\ \left|d\vec{F}_{fr}(\theta, z)\right| &= K_{fr}(z)dA_r(\theta, z) \end{aligned} \tag{5.19}$$

where K_{nr} and K_{fr} are cutting coefficients which can predict the local relationship of differential forces to the contact area in the axial direction

$$\begin{aligned} K_{nr}(z) &= (c_3 + c_4 z + c_5 z^2) \\ K_{fr}(z) &= (c_6 + c_7 z + c_8 z^2)K_{nr}(z) \end{aligned} \tag{5.20}$$

The differential normal and frictional edge forces , $d\overrightarrow{F}_{ne}$ and $d\overrightarrow{F}_{fe}$, are proportional to the differential cutting edge length ds

$$\left|d\overrightarrow{F}_{ne}(\theta,z)\right| = K_{ne}(z)ds(z)$$

$$\left|d\overrightarrow{F}_{fe}(\theta,z)\right| = K_{fe}(z)ds(z) \tag{5.21}$$

where K_{ne} and K_{fe} are also assumed to have same local relationship with the differential edge length in the axial direction

$$K_{ne}(z) = (c_9 + c_{10}z + c_{11}z^2)$$

$$K_{fe}(z) = (c_{12} + c_{13}z + c_{14}z^2)K_{ne}(z) \tag{5.22}$$

Finally, the instantaneous cutting force acting on the cutting edge can be expressed as follows:

$$\overrightarrow{F}_{pr}(\theta) = \int_{z_{in}}^{z_{out}} [K_{nr}(z)\hat{n}_r(\theta,z) + K_{fr}(z)\hat{n}_{fr}(\eta_c,\theta,z)]dA_r(\theta,z) + \tag{5.23}$$

$$[K_{ne}(z)\hat{n}_e(\theta,z) + K_{fe}(z)\hat{n}_{fe}(\theta)]ds(z) \tag{5.24}$$

The integration limits (z_{in} and z_{out}) are extracted using the geometric simulation module.

5.3.2 Model calibration

The calibration method to determine the cutting coefficient vector $\overrightarrow{C}(c_1, c_2, ..., c_{14})$ is implemented here for the half-immersion down-milling case while spindle speed (N)

and feed (F_t) are 450 rpm and 0.1 $mm/tooth$, respectively. The integration limits for half-immersion are depicted in Figure 5.17. A constrained nonlinear optimization based on Sequential Quadratic Programming (SQP) is used here to minimize the sum of square errors (SSE) between the predicted and measured force components.

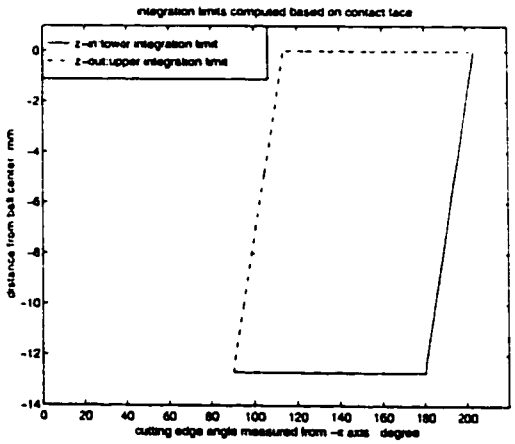


Figure 5.17: Half-immersion integration limits

$$\min SSE(\vec{C}) = \sum_{i=1}^{M} \left| \vec{F}_{pr}(\theta_i, \vec{C}) - \vec{F}_{me}(\theta_i) \right|^2 \tag{5.25}$$

where M and F_{me} are the number of measurements and the measured cutting forces, respectively.

Figure 5.18 shows the measured components of the cutting forces for the half-immersion case and the prediction of the calibrated model (solid line). The agreement between the measured and the predicted forces are excellent. Small discrepancies can be the result of unmodeled rubbing effects in the vicinity of the ball-nose tip, a slight

shift in the reference of measurement, and also small differences between geometric representations and the true cutting edge and rake face.

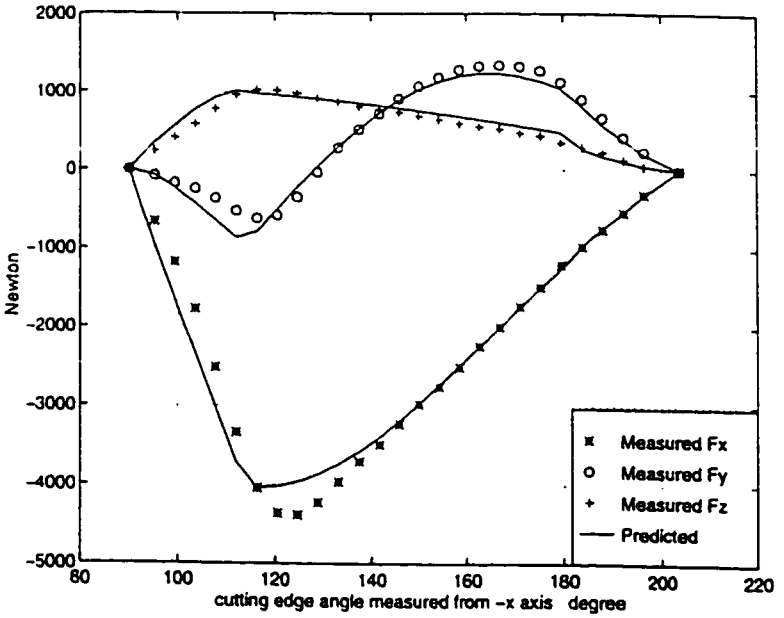
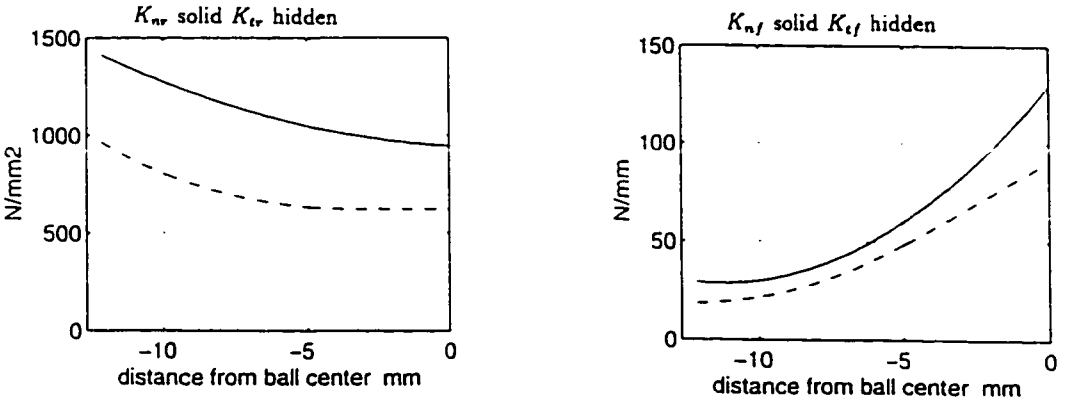


Figure 5.18: Model calibration

The variations of cutting coefficients on the rake face (K_{nr} and K_{fr}) and on the edge (K_{ne} and K_{fe}) are depicted in Figure 5.19. On the rake face the cutting coefficients are monotonically increasing with axial direction. This is mainly due to the decrease in the radial rake angle which cause the increase in the pressure on the rake face. In addition, low cutting speeds in the vicinity of the tool tip can lead to higher cutting coefficients on the rake face. On the contrary, the edge cutting coefficients are monotonically decreasing with axial direction. As a result of the tool deflection, the magnitude of normal edge force decrease as it reaches to the tool tip. The chip flow angle (η_c) is approximately equal to the helix at the starting point and decreases as it reaches to the tool tip (Figure 5.20).



a- coefficients on the rake face b- coefficients on the cutting edge

Figure 5.19: Variations of cutting coefficients

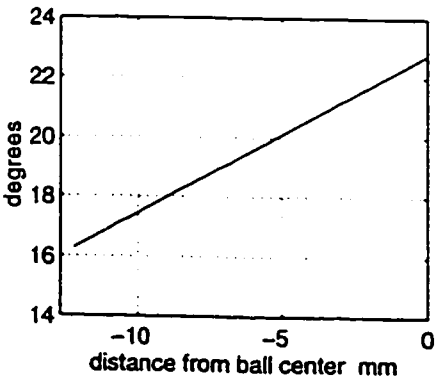


Figure 5.20: Variation of chip flow angle

5.4 Experimentation and verification

Experimental cutting tests were performed on a three-axis CNC machining center (see Section 4.4.1). This machine tool is relatively rigid and no significant vibrations were experienced during the cutting tests. A Kistler type 9255A, three-force component table dynamometer was used to measure F_x, F_y, and F_z components of the instantaneous cutting force. The dynamometer was quite stiff (natural frequency is 2000Hz). A spindle encoder was also used as an external clock to synchronize sampling with the cutting edge position.

Three-axis ball-end milling with different immersion geometries were performed on workpieces of steel 1018. The cutters used were sharp one inch diameter HSS ball-end mills to minimize the effects of the run-out and wear. The milling operations were performed in the down-milling mode while the spindle speed and feed set to 450 *rpm* and 0.1 *mm/tooth*, respectively. The immersion geometry as well as the in-cut segments of the cutting edge are depicted in Figure 5.21. Also, the integration limits for uphill angles 10, 15, 20, and 25 degrees are shown in Figure 5.22.

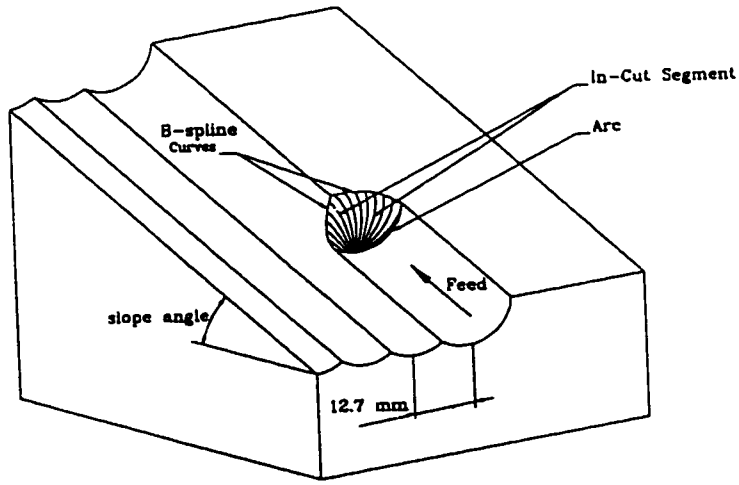


Figure 5.21: Three-axis immersion geometry and in-cut segments

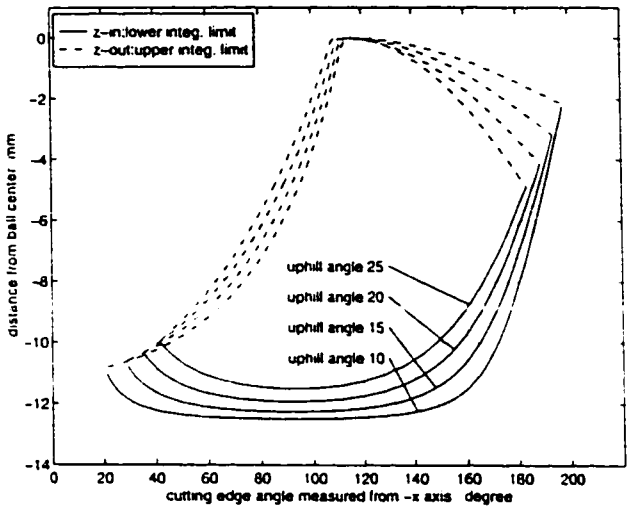


Figure 5.22: Integration limits of three-axis ball-end milling

Results and Discussions

Although the developed force model was calibrated for half-immersion ball-end milling, it can accurately predict three components of instantaneous cutting forces for various two- or three-axis ball-end milling with different immersion geometries. The measured and predicted cutting forces for uphill angles of 10, 15, 20, and 25 degrees are depicted in Figures 5.23 and 5.24. There is an excellent agreement between measurements and predictions. The small discrepancy in predicting peak values may be due to a slight shift in the reference of measurement, the small difference between geometric representations and the true geometries and the run-out of the cutting edges. The amount of the overestimation of the maximum resultant force increases with the uphill angle. This is due to the fact that the calibration was performed for the half-immersion case in which the tool tip is completely engaged. Thus, as a result of unmodeled rubbing effects of the tool tip, an overestimation is expected for the cases in which the tool tip is not engaged.

It has also been observed that as the uphill angle increases the magnitude of

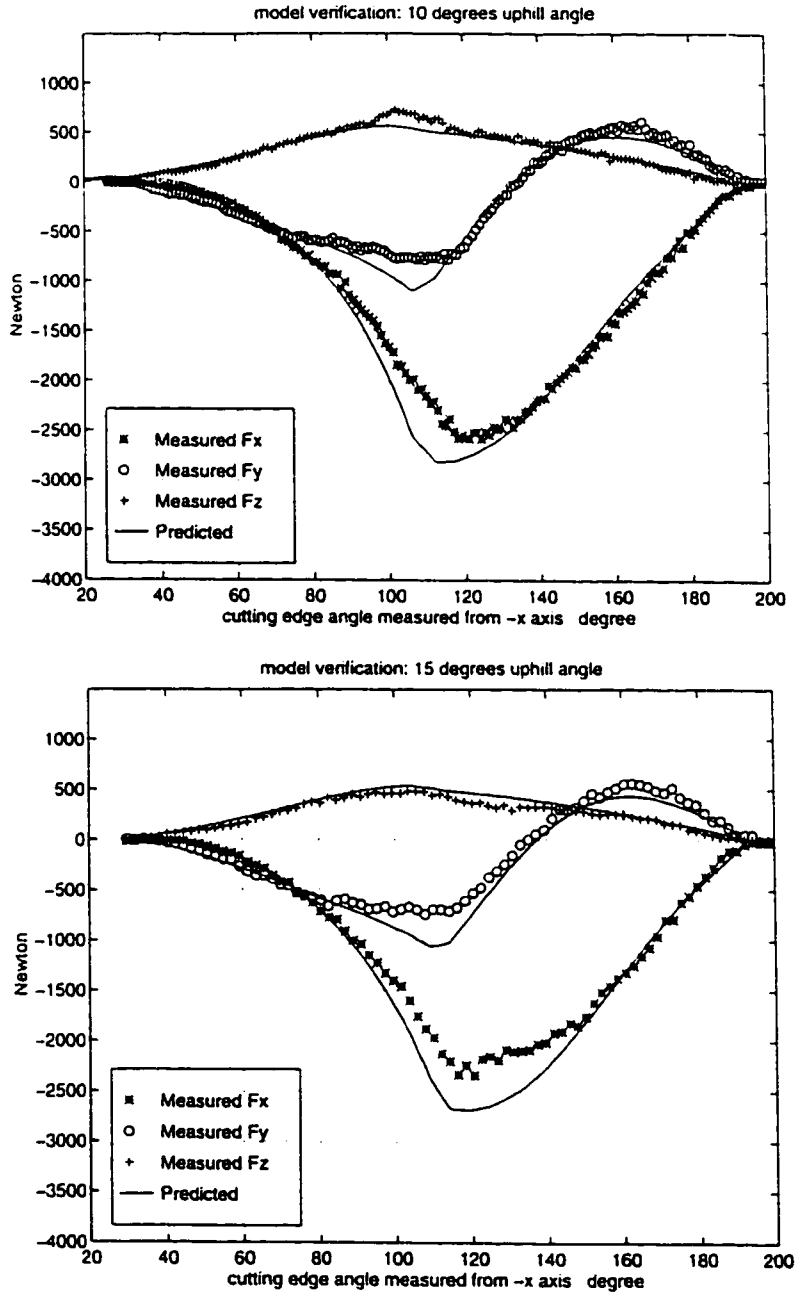


Figure 5.23: Predicted and measured cutting forces for 10 and 15 degrees uphill angles

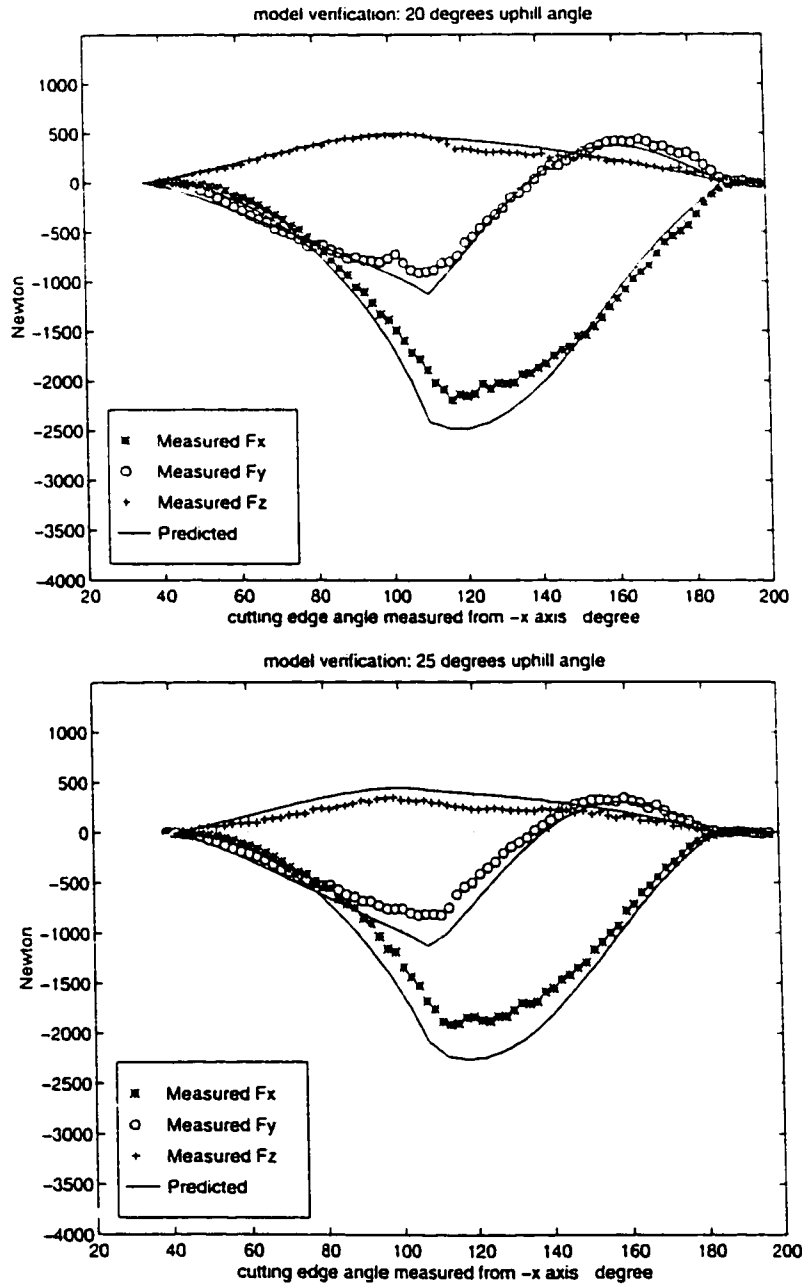


Figure 5.24: Predicted and measured cutting forces for 20 and 25 degrees uphill angles

resultant cutting forces significantly decreases which suggests that cutting within the vicinity of the ball-nose tip should be avoided. Practically, this can be achieved by tilting the tool. The main disadvantage of tilting the ball-end mill is the added complexity in terms of NC codes and machine tool requirements. However, this action can be justified as it can promote longer tool life, lower cutting forces and lower cutting power.

5.5 Feed scheduling

For the semi-finishing operation, the dominant constraints are the tool deflections over the whole length of the tool paths. The tool deflection normal-to-the-path increases the amount of chip load left for finishing operation while the tool deflection tangent-to-the-path can cause serious surface gouging problems [16][38][39]. The stability constraint can be ignored in the semi-finishing operation of the typical die material in the annealed state.

The cutting simulation system developed first predicts the maximum cutting force based on the extracted chip geometry. Then, the tool deflections (δ_x and δ_y) at each tool position are estimated using the static stiffness of the structure along x and y directions at the tool tip (k_x and k_y). The x and y deflections are decomposed into the tangent- and normal-to-the-path directions. The feed along the contour path is adjusted based on the tangent deflection constraint in order to avoid surface gouging. The maximum allowed tangent deflection depends on the tool path curvature (or the design surface geometry) [38]. For example, in the corner areas where the curvature is large a tighter defection constraint is required.

5.6 Summary

The ball-end milling process simulation methodology has been developed and implemented. The simulation system consists of geometric and physical simulation modules. The geometric module can accurately simulate two- and three-axis ball-end milling operations and update the in-process part. The developed geometric simulator is based on B-rep NURBS enhanced solid modeling techniques. It can also automatically extract the chip geometry (in-cut segments and undeformed chip thickness) for general two- and three-axis ball-end milling operations. The B-spline representation of the cutting edge also makes it possible to be used for any ball-end mill geometry.

The physical module consists of a newly developed force model which can accurately predict three components of the cutting force. This force model utilizes the chip geometry extracted by the geometric simulator to estimate three components of the instantaneous cutting forces for various uphill angles and immersion geometries. The force model has been verified in different 3-axis ball-end milling experiments. The predicted cutting forces can be used in feed scheduling subjected to the deflection constraints. Experimental results confirm the validity of the developed system.

Chapter 6

Finishing operation

The geometric modeling issues of the finishing operation with ball-nosed cutter are discussed in this chapter. In the finishing operation, the surface texture defined in terms of roughness, waviness, lay and flaws ([3]), deteriorates due to several factors such as built-up edge formation, dynamics of structure, and periodic engagements of the cutting edge. In the first section of this chapter, the procedure of updating the B-rep model of the part for the finishing operation is developed. The method is capable of precisely constructing the feed mark geometry as well as the cross-feed scallop geometry. In the next section, the feed mark profile is extracted from the B-rep model and compared with the experimental measurements. Also, the effects of feed and tool diameter variations on the peak-to-valley maximum height (R_{max}) are investigated in the following section. The last section summarizes the applications of the methodology developed.

6.1 B-rep model of feed marks and scallops

In the finishing operation of sculptured surfaces, ball-nose cutters are used extensively. For these cutters, the inserts are designed to produce a radius within a certain tolerance on the workpiece. The cutting edge of the insert used in the experiment is one quarter of a circle which is tilted by three degrees, (Figure 6.1). The surface swept by the cutting edge is constructed through skinning n instances of the cutting edge, which are transformed according to true cutting motions of the edge. In addi-

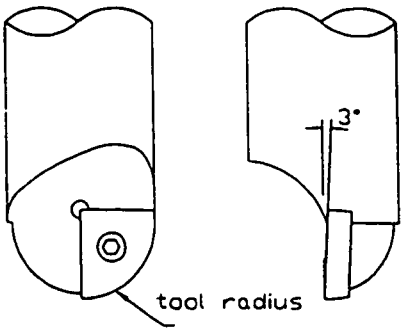


Figure 6.1: Insert geometry

tion to the primary cutting and translational feed motions, the tool run-outs and tool deflections influence the cutting edge position [21]. Since cutting forces are relatively low in finishing, the contribution of deflections is ignored in the current work. Thus, the ith transformation matrix corresponding to the edge angular position θ_i can be expressed by

$$\mathbf{Tr}(\theta_i) = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & \mathrm{e}\,\sin(\theta - \theta_\mathrm{e}) \\ \sin(\theta_i) & \cos(\theta_i) & 0 & \mathrm{f_{ty}} + \mathrm{e}\,\cos(\theta - \theta_\mathrm{e}) \\ 0 & 0 & 1 & \mathrm{f_{tz}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.1}$$

assuming that the feed motion lies in yz plane and is given by

$$\begin{aligned} f_{ty} &= \frac{f_t \cos(\phi)\,\theta_i}{360} \\ f_{tz} &= \frac{f_t \sin(\phi)\,\theta_i}{360} \end{aligned} \tag{6.2}$$

where ϕ is the uphill angle. The transformation matrix also includes the tool radial run-out e which has θ_e degrees lag with respect to the cutting edge. Figure 6.2 shows the skinned surface generated for the total rotation angle of 183 degrees where the cutting takes place. The extra three degrees compensate for the cutting edge tilt angle. To avoid the surface degeneration at the ball-nose tip, the area within five degrees to the vertical axis is omitted.

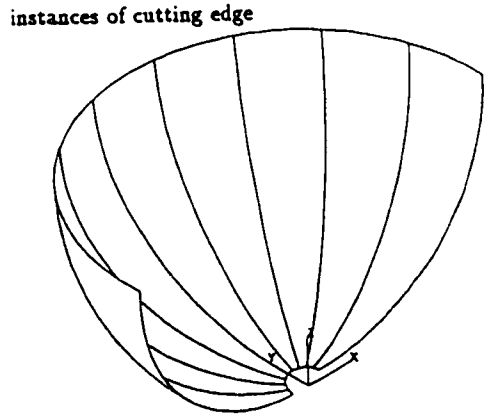


Figure 6.2: Surface swept by the cutting edge with coupled cutting and three-axis feed motions

In the next step the sheet body of the skinned surface is generated. Then, a copy of the sheet body for each tool rotation is generated and transformed along the tool path. The spacing between copies is the feed f_t in the case of three-axis milling.

Thus, the material removal process of ball-end milling can be precisely simulated by performing consecutive Boolean subtractions between the part and the transformed copies of the sheet body. Figure 6.3 illustrates the final B-rep model of the workpiece which includes feed mark and cross-feed scallop geometries. Although the proposed method is implemented for the 3-axis milling operation, it can be extended to four- and five-axis milling by modifying the transformation matrix (Equation 6.1).

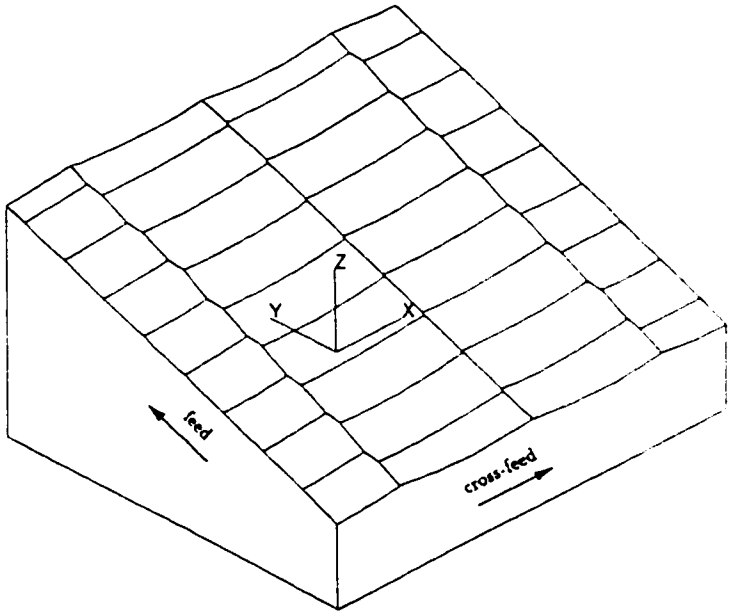


Figure 6.3: B-rep model of feed marks and cross feed scallops

6.2 Feed mark profile

The faces readily available in the B-rep model precisely represent the feed mark geometry on the surface. To obtain the feed mark profile, these faces can be intersected with a vertical plane aligned with feed direction. The resulting intersection curves represent the feed mark profile.

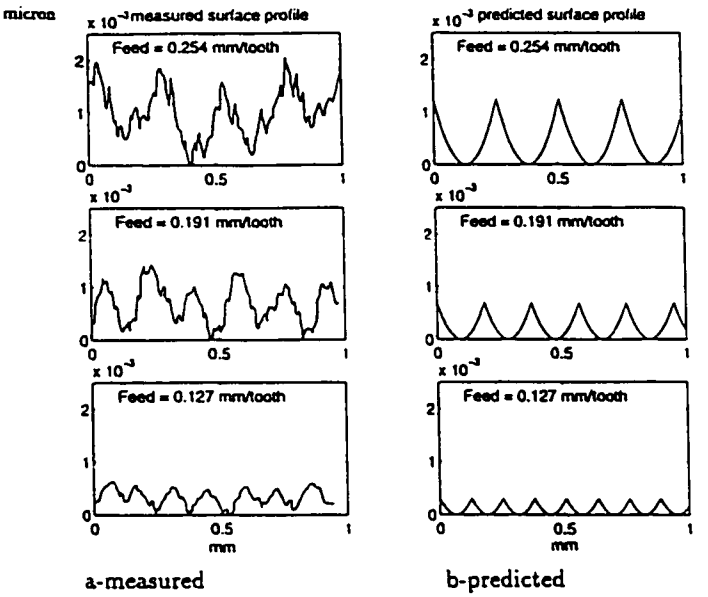
The cutting conditions used for the simulation and experimentation are as follows:

Down-milling with a 12.7 mm diameter ball-end mill is performed with 1.59 mm radial width of cut. The depth of cut is maintained at 1.27 mm while cutting the uphill angle of 20 degrees. The feeds are 0.127, 0.191, and 0.254 mm/rev with a cutting speed of 10,000 RPM. Also, the radial run-out measured for the experimental setup was found to be 0.012 mm with 40 degrees lag with respect to the cutting edge.

Figure 6.4 depicts the measured feed mark profile using a stylus contact-type profilometer. The spacing and pick-to-valley values of the measured profile are in reasonable agreement with the predictions which validate the accuracy of the geometric simulation. Small discrepancies in the pick-to-valley values may be due to the fact that in the experimental setup, there was an inevitable small misalignment between the arm of the profilometer and the feed direction. The important parameters in ball-end milling which influence the feed mark profile are feed and tool diameter while the uphill angle does not have much effects [32]. The surface representing R_{max} variations with respect to feed and tool diameter is illustrated in Figure 6.5 for the range used in practice. It has been observed that R_{max} gradually decreases with the tool diameter while it sharply increases with the feed.

6.3 Summary

The finishing material removal is simulated based on the true path of each point on the cutting edge. Thus, it is possible to precisely construct the feed mark and scallop geometries on the surface. Experimental measurements confirm the validity of the implemented methodology. The system developed can be integrated with CAD/CAM systems to predict the surface texture for the finishing operation of sculptured surfaces



a-measured b-predicted

Figure 6.4: Surface roughness profile

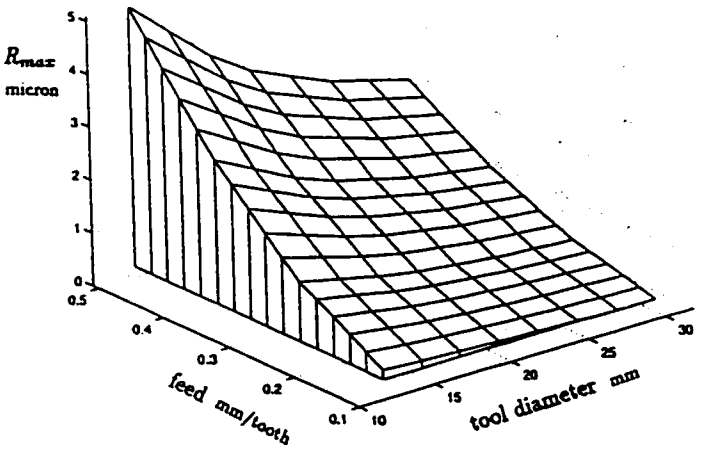


Figure 6.5: Variation of peak-to-valley R_{max} with feed and tool diameter

as well as the finishing chip load.

Chapter 7

Summary

Geometric modeling techniques, in particular B-rep solid modeling techniques, have become available for many of today's CAD/CAM systems. The state of the art CAM systems are capable of generating tool paths for sculptured surfaces with prescribed tolerances; however, their role in achieving reliable and efficient NC codes is still limited. They provide little assistance to the part programmer in terms of choosing the proper cutting conditions. The NC code verification performed by these systems is limited to checking the nominal geometric dimensions. Also, they do not take the physical/mechanical aspects of the machining process into consideration.

The comprehensive and realistic prediction of the physical/mechanical parameters requires a model-based environment with the functions of physical, geometrical, and interaction modeling.

This thesis has undertaken to overcome the above limitations by developing a model-based machining simulation methodology which is implemented for the different stages of die cavity milling operations, that is roughing, semi-finishing, and

finishing. The implemented methodology is twofold. The first part is the geometric modeling of the milling operations with varying levels of accuracy and complexity. The second part is the physical/mechanical simulation which takes into account the mechanics of the cutting process as well as the dynamics of the tool and structure. To achieve a comprehensive and efficient milling simulation, the critical interactions between the geometric and the physical/mechanical models are also simulated.

The list of contributions achieved in the course of the current research are recounted as follows:

- The pocketing simulation methodology has been developed for the die cavity roughing operation and implemented in the form of a model-based pocketing simulation system.

    1. The geometric simulation module constructs the volume swept by the cutter, accurately updates the B-rep model of the part as the cutter removes the material from the stock, and automatically extracts the immersion geometry from the B-rep updated model.

    2. The physical simulation module, integrated with the geometric module, is a regenerative mechanistic force model which simulates the closed loop of the machine tool-cutting process. It accurately predicts the instantaneous cutting forces as well as maximum tool tip deflections. The physical module can also anticipate the onset of chatter vibrations.

    3. The applications of the developed system in the feed scheduling of the contour path and in the prediction of the most stable direction of the machining unit are discussed.

- The semi-finishing simulation methodology has been also developed. The simulation system is successfully implemented for two- and three-axis ball-end milling operations.

    1. Based on advanced sweeping/skinning techniques, the geometric module constructs the volume swept by the the ball-end cutters for two- and three-axis tool paths. It precisely updates the B-rep model of the in-process part and automatically extracts the chip geometry (in-cut segments and undeformed chip thickness) for general two- and three-axis ball-end milling operations. The NURBS implementation makes it possible for the method to be used for any form cutters, e.g. taper and bull-nose cutters.
    2. The physical module consists of a newly developed force model which accurately predicts three components of the cutting force. This force model utilizes the chip geometry extracted by the geometric simulator to estimate three components of the instantaneous cutting forces.
    3. The physical simulation module has been verified in different three-axis ball-end milling experiments for various uphill angles and immersion geometries.
    4. The applications of the developed system in feed scheduling subjected to the deflection constraints are discussed
    5. Based on the knowledge of the material remaining on the surface, represented by the updated B-rep model, two- and three-axis tool paths are computed. In the semi-finishing operation, a three-axis milling strategy is developed to control the scallop height remaining on the surface and avoid

cutting with the dead zone of the cutter.

- The finishing material removal is simulated by perform Boolean subtract operations between the updated B-rep model of the part and the surface swept by the cutting edge. The swept surface is computed based on the true path of each point on the cutting edge.

  1. The method precisely constructs the feed mark and scallop geometries remaining on the surface and efficiently extracts the surface profile and scallop height from the B-rep model. Experimental measurements confirm the validity of the implemented methodology.
  2. The developed system can be integrated with CAD/CAM systems to predict the surface texture for the finishing operation and to compute the finishing chip load.

Further research efforts can be followed in the areas listed below:

- The methodology implemented for die cavity milling operations can be extended to model some other equi-technological free-form surfaces. The main aim here is to investigate the most efficient strategy to perform milling operations on the equi-technological surface found in the die or mold.

- Additional research work is required to extend the developed three-axis swept volume generation technique to a four- or five-axis technique. In this regard, issues related to surface degeneration must be taken into account.

- A great deal of effort is required for five-axis finishing tool path planning based on the knowledge of material remaining on the surface after semi-finishing. In

this regard, the main issue is the tool wear prediction along the cutting edge. Also, research effort is required to model the chip load as well as the feed mark, and scallop geometries for the five-axis finishing operation.

- A five-axis mechanistic force model is needed to predict the components of the cutting force based on the extracted chip load. In addition to the spindle and tool dynamics, the dynamic behavior of the rotary tables should be taken into consideration.

Appendix A

Tolerances in the B-rep geometric modeler

Numbers in computer are represented by floating-points. Since a floating-point number consists of an exponent and a mantissa, which are fixed length integers, there is a compromise between the precision (i.e., the number of decimal places) and the range (i.e., the difference between the largest and smallest number representable). The ratio between the largest and smallest values representable in **ACIS** is known as **resnor** and is equal to 10^{-10}. This ratio represents the number of digits available to store a number. The smallest number representable in **ACIS** is known as **resabs** which is the distance below which the software considers two points to be coincident. The default value for **resabs** is 10^{-6}. With these two values and assuming that the unit is one millimeter, the default modeling world (Figure A.1) is

$$10^{-6}\text{mm} \Longleftrightarrow 10^{4}\text{mm or}$$

$$1^{-3}\text{ micron} \Longleftrightarrow 10\text{ meter}$$

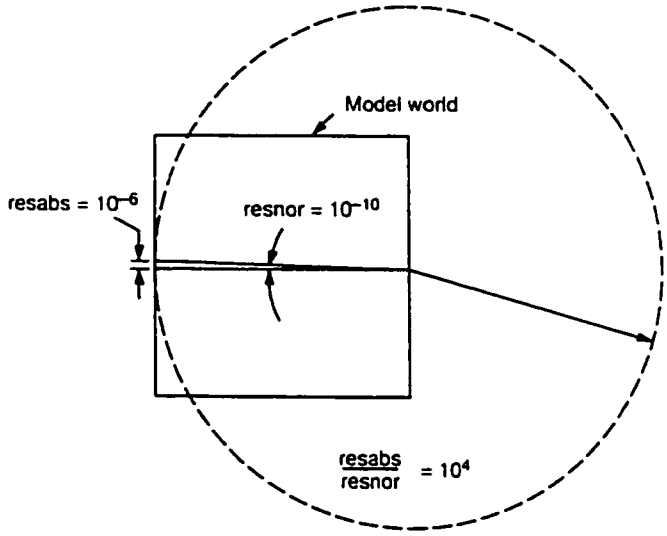


Figure A.1: Modeling world precision and range

The **resabs** value is used for a wide range of geometrical classifications and checks; for example, classifying a point relative to an object or checking the model geometrical integrity. The model integrity is checked as follows. The surface of the FACEs must intersect along the curves of the EDGEs on their common boundary within the specified tolerance. Similarly, the points of the VERTEXs must lie on the curves of the EDGEs, again within the specified tolerance.

The tolerance to which a NURBS curve or surface is fit to an exact geometry or a given set of data points is controlled by **resfit** value in **ACIS**. This tolerance is used in all spline related geometric operations. For example, in the construction of the three-axis swept volume **ACIS** spline functions approximate a set of given curves placed along the tool path.

The surface approximation error depends on the **resfit** value and the number of section curves. The entities generated by the default value of **resfit** (i.e., 10^{-3}mm)

and 100 section curves hold large approximation errors. Consequently, the operation performed on these entities results in invalid objects. For instance, the swept volume generated to update the model of the die shown in Figure 5.10 is constructed by stitching ten FACEs. The edges of FACE 1 to FACE 4 (Figure 5.7) are spline curves with large fitting tolerances; therefore, the constructed swept volume BODY is invalid (invalid geometric integrity: 8 coedges without a partner on a single-sided FACE).

With the limitations on the heap space and to avoid building a very large .sat file (**ACIS** file format), it is not recommended either to increase the number of section curves or to tighten the fitting tolerance. The alternative approach which is proposed and implemented by the current research is to use advanced sweeping/skinning techniques to construct the tool path dependent FACEs of the swept volume (FACEs 1 to 2 in Figure 5.7).

Figure A.2 compares the approximation/fitting errors for the section curve of FACE 1 which is a semi-circle. The dashed curve represents the fitting error of the cubic B-spline generated by **ACIS** when sweeping a semi-circle along the tool path. The spline fitting tolerance (**resfit**) is set to the default value (10^{-3}mm). The maximum fitting error is almost twice the default value. The solid curve represents the fitting error of the quadratic B-spline curve computed by the method proposed in Section 5.2.1. Although both of the methods have the same number of control points and knots, the fitting error of the latter is improved to 0.2 micron.

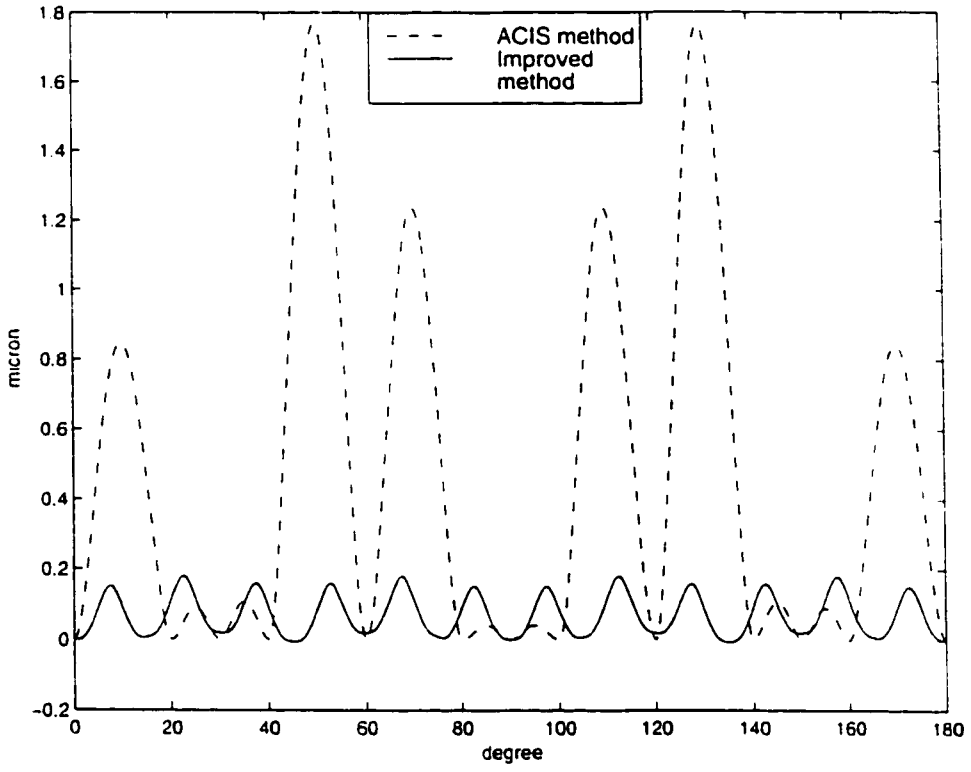


Figure A.2: Improving fitting tolerance in interpolating a semi-circle

Appendix B

Modal testing

Modal testing is an experimental technique used to derive a mathematical description of the dynamic characteristics of a linear vibratory system. This mathematical description usually comprises the modal data of the system such as natural frequencies, mode shapes and damping factors. The technique is based on the derivation of the relationship between the vibration response at one degree of freedom (DOF) and excitation at the same or another degree of freedom as a function of excitation frequency. This relationship is known as the frequency response function(FRF) which is often a complex mathematical function. Combinations of excitation and response at different degrees of freedom lead to a complete set of frequency response functions which can be collectively represented by an FRF matrix of the system. This matrix is usually symmetrical, reflecting the structural reciprocity of the system.

The practice of modal testing involves measuring the frequency response functions of a dynamic system. This can be simply done by applying a measured excitation

at one degree of freedom of the system in the absence of other excitations and measuring the vibration response at one or more degrees of freedom. Modern excitation techniques and recent development of modal analysis theory permit more complicated excitation mechanisms. The excitation can be of a selected frequency band, stepped sinusoid, transient or white noise. It is usually measured by *a force transducer* at the driving point while the response by is measured by *an accelerometer*. Both the excitation and response signals are fed into a dual channel fast Fourier Transform (FFT) analyzer which computes the frequency response function.

A practical consideration of modal testing is how many frequency response functions need be measured in order to adequately derive the dynamic characteristics of the tested object. Theoretically, either one row or one column of the FRF matrix is needed to derive a modal description of a system (n measurements, where n is the number of DOF). However, in order to determine the vibration of the system in the local coordinates, an additional $\frac{n(n+1)}{2}$ measurements should be made. When sufficient FRF measurements have been accomplished, numerical analysis will be conducted to derive modal parameters from the measured FRF data by ways of curve fitting.

Modal parameters M, C, K, and w_{n_i} of the tool-spindle assembly can be obtained from one measurement of $X(\omega)/F(\omega)$ at a significant point (e.g. the tool tip). The transfer function (TF) can be described by either magnitude and phase or real and imaginary parts. Since there is small damping in the modes interested (tool and spindle modes), either the real or the imaginary part of TF is sufficient. The real part of direct TF at the tool tip for the x direction is given by [63]:

$$G_{jj}(\omega) = \frac{X_j(\omega)}{F_j(\omega)} = \sum_{i=1}^{n} \frac{1}{k_{q_i}} \frac{{\omega_{n_i}}^2({\omega_{n_i}}^2 - \omega^2)}{({\omega_{n_i}}^2 - \omega^2)^2 + 4\eta^2{\omega_{n_i}}^2\omega^2} \tag{B.1}$$

The measured TF is the sum of the individual normal TF at the measured point. The general form of the TF real part and its components are shown in Figure B.1. The peak values of each component are as follows :

$$G_{max_i} = \frac{1}{k_{q_i}} \frac{1}{4\eta_i(1-\eta_i)} \tag{B.2}$$

$$G_{min_i} = -\frac{1}{k_{q_i}} \frac{1}{4\eta_i(1+\eta_i)} \tag{B.3}$$

At the natural frequency ω_{n_i}, G is zero. The damping ratio can be approximately computed by:

$$\eta_i = \frac{\Delta\omega}{2\omega_{n_i}} \tag{B.4}$$

where $\Delta\omega$ is the difference in the frequencies of two peaks.

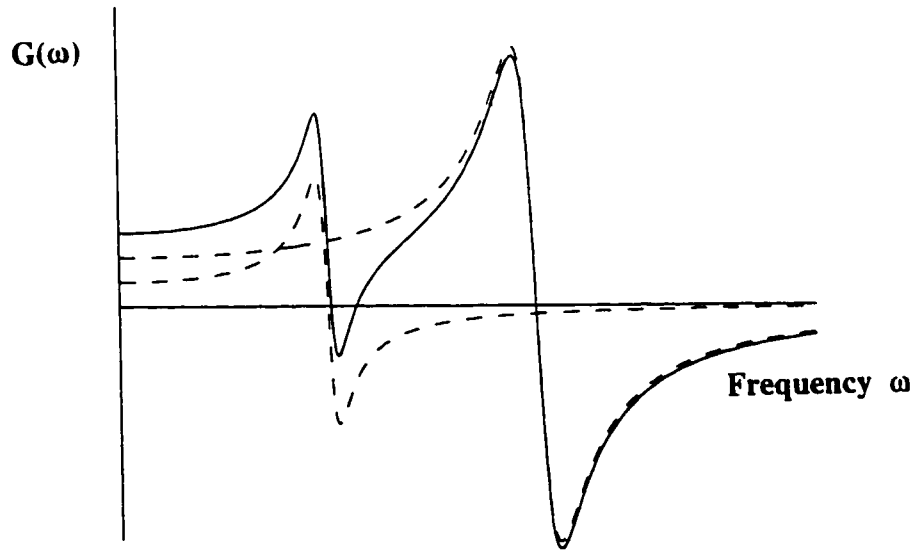


Figure B.1: General form of measured transfer function: real part

The measured TF is decomposed according to Equation B.1 by curve fitting and

all n values of m_{q_i}, c_{q_i}, and k_{q_i} are computed. The following table summarizes the results of the modal testing performed on the setup shown in Figure 4.21. In order

		Tool mode	Spindle mode
ω_{nx}	[Hz]	1898	968
k_{qx}	[N/m]	7.25e7	2.74e8
η_{qx}	--	0.040	0.031
m_{qx}	[kg]	0.51	7.41
ω_{ny}	[Hz]	1909	767
k_{qy}	[N/m]	8.86e7	1.61e8
η_{qy}	--	0.036	0.074
m_{qy}	[kg]	0.62	6.93

Table B.1: Modal parameters of tool-spindle assembly

to perform the testing, the flat-end mill was replaced with a mandrel of the same length and diameter. The method of excitation was the application of an impact using a Kistler hammer type 9726A which was also able to measured the excitation force. The response of the setup was measured at the tool tip using a Kistler Piezo-electric accelerometer type 8702B25. An HP frequency analyzer was used to collect the data and compute the frequency response of the assembly. The average of 10 measurements, performed for x- and y-directions, was computed while the coherence was kept in the range of 0.9 to 1.0.

References

[1] F. Abrari and M.A. Elbestawi. Closed form formulation of cutting forces for ball and flat end mills. *International Journal of Machine Tool Design and Research*, 37(1):17–27, 1996.

[2] Y. Altintaş and A.D. Spence. End milling force algorithms for CAD systems. *CIRP Annals*, 40(1):31–34, 1991.

[3] American National Standards Institute, Inc., 11 West 42nd Street, New York, NY, USA 10036. *Surface Texture (Surface Roughness, Waviness and Lay)*, 1992.

[4] E.J.A. Armarego and R.H. Brown. *The Machining of Metals*. Prentice Hall, 1969.

[5] E.J.A. Armarego and N.P. Deshpande. Force prediction models and CAD/CAM software for helical tooth milling processes, part iii, end-milling and sloting operations. *International Journal of Production Research*, 31, 1993.

[6] E.J.A. Armarego and N.P. Deshpande. Force prediction models and CAD/CAM software for helical tooth milling processes, part i, basic approach and cutting analyses. *International Journal of Production Research*, 31(8):1991–2009, 1993.

[7] E.J.A. Armarego and N.P. Deshpande. Force prediction models and CAD/CAM software for helical tooth milling processes, part ii, peripheral milling operations. *International Journal of Production Research*, 31(10):2319–2336, 1993.

[8] E.J.A. Armarego, A.J.R. Smith, and Z.J. Gong. Four plane facet point drills - basic design and cutting model predictions. *CIRP Annals*, 39(1):41–45, 1990.

[9] E.J.A. Armarego and R.C. Witfield. Computer based modeling of popular machining operations for force and power predictions. *CIRP Annals*, 34(1):65–69, 1985.

[10] T. Bailey, Y. Ruget, A. Spence, and M.A. Elbestawi. Open-architecture controller for die and mold machining. In *Proceedings of the 1995 American Control Conference*, pages 200–204, Seatle, Washington, USA, June 1995. American Automatic Control Council, IEEE.

[11] J.S. Bendat and A.G. Piersol. *Engineering Applications of Correlation and Spectral Analysis*. John Wiley and Sons, 1980.

[12] P. Bertok, S. Takata, K. Matsushima, J. Ootsuka, and T. Sata. A system for monitoring the machining operation by referring to a predicted torque pattern. *CIRP Annals*, 32:439–442, 1983.

[13] A. Bhattacharyya. *Metal Cutting: Theory and Practice*. Central Book Publishers, India, 1984.

[14] E. Budak and Y. Altintas. Prediction of milling force coefficients from orthogonal cutting data. *ASME Journal of Manufacturing Science and Engineering*, 64:453–460, 1993.

[15] E. Budak, Y. Altintas, and E.J.A. Armarego. Prediction of milling force coefficients from orthogonal cutting data. *ASME Journal of Manufacturing Science and Engineering*, 118:216–224, 1996.

[16] B.K. Choi. *Surface Modeling for CAD/CAM.* Elsevier Science, 1995.

[17] Z Dong, H. Li, and G.W. Vickers. Optimal rough machining of sculptured parts on a cnc milling machine. *ASME Journal of Engineering for Industry*, 115:424–431, 1993.

[18] D. Dragomatz and S. Mann. A classified bibliography of literature on NC milling path generation. *Computer-Aided Design*, 29(3):239–247, 1997.

[19] H. El-Mounayri, A.D. Spence, and M.A. Elbestawi. Enhanced CAD/CAM for simulation and optimization of 3-5 axis milling of dies and molds. In *Proceedings of the CSME 13th Sym. On Eng. Applications of Mechanics: Manufacturing Science and Engineering*, pages 394–401, McMaster University, Hamilton, Canada, May 1996.

[20] M.A. Elbestawi, F. Ismail, R. Du, and B.C. Ullagaddi. Modelling machining dynamics including damping in the tool-workpiece interface. *ASME Journal of Engineering for Industry*, 116(4):435–439, 1995.

[21] M.A. Elbestawi, F. Ismail, and K.M. Yuen. Surface topography characterization in finish milling. *International Journal of Machine Tool Design and Research*, 34(4):245–255, 1994.

[22] G.E. Farin. *Curves and Surfaces for Computer Aided Geometric Design A Practical Guide.* Academic Press, Inc., San Diego, CA, third edition, 1993.

[23] H. Feng and C. Menq. The prediction of cutting force in ball-end milling process - part 1 : Model formulation and model building procedure. *International Journal of Machine Tool Design and Research*, 34(5):697–710, 1994.

[24] H. Feng and C. Menq. The prediction of cutting force in ball-end milling process - part 2 : Cut geometry analysis and model verification. *International Journal of Machine Tool Design and Research*, 34(5):711–719, 1994.

[25] H. Feng and C. Menq. A flexible ball-end milling system model for cutting force and machining error prediction. *ASME Journal of Manufacturing Science and Engineering*, 118:461–469, 1996.

[26] J.C.E. Ferreira and S. Hinduja. A radial tool path generation method for pocket-machining. In *Proceedings of the North American Manufacturing Research Conference*, volume XXI, pages 263–270, 1993.

[27] M. Field, J.F. Kahles, and W.P. Koster. *Metals Handbook(machining): Surface Finish and Surface Integrity*, volume 16. ASM International, ninth edition, 1989.

[28] M. Held. Geopocket- a sophisticated computational geometry solution of geometrical and technological problems arising from pocket machining. In *cape 89*, pages 283–293, 1989.

[29] M. Held, G. LuKacs, and L. Andor. Pocket machining based on contour parallel tool paths generated by means of proximity map. *Computer-Aided Design*, 26(3):189–203, 1994.

[30] C.M. Hoffman. *Geometric and Solid Modeling–An Introduction.* Morgan Kaufmann, San Mateo, CA, 1989.

[31] Y.C. Huang and J.H. Oliver. Integrated simulation, error assessment, and tool path correction for five-axis nc milling. *Journal of Manufacturing Systems*, 14(5):331–344, 1995.

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

[33] W.A. Kline, R.E. DeVor, and R. Lindberg. The prediction of cutting forces in end milling with application to cornering cuts. *International Journal of Machine Tool Design and Research*, 22(1):7–22, 1982.

[34] F. Koenigsbereger and J. Tlusty. *Machine Tool Structures*, volume 1. Pergamon Press, 1970.

[35] D.E. LaCourse. *Handbook of Solid Modeling*. McGraw-Hill, Inc., 1994.

[36] E.H. Lee and B.W Shaffer. Theory of plasticity applied to a problem of machining. *ASME Journal of Applied Mechanics*, 18(4):405, 1951.

[37] Y.S. Lee and T.C. Chang. Cascam- an automated system for sculptured surface cavity machining. *Computers in Industry*, 16:321–342, 1991.

[38] E.M. Lim, H. Feng, C. Menq, and Z. Lin. The prediction of dimensional error for sculptures surface productions using the ball-end milling process. part 1: Chip geometry analysis and cutting force prediction. *International Journal of Machine Tool Design and Research*, 35(8):1149–1169, 1995.

[39] E.M. Lim, H. Feng, C. Menq, and Z. Lin. The prediction of dimensional error for sculptures surface productions using the ball-end milling process. part 2: Surface generation model and experimental verification. *International Journal of Machine Tool Design and Research*, 35(8):1171–1185, 1995.

[40] M.E. Martellotti. An analysis of the milling process. *Transactions of the ASME*, 63:677–700, 1941.

[41] J.P. Menon and D.M. Robinson. High performance nc verification via massively raycasting: Extension to new phenomena and geometric domains. In *Proceedings of the 1991 ASME Winter Annual Meetings, 8-13 Nov., Anaheim, California*, pages 179–194. ASME, 1992.

[42] J.P. Menon and H.B. Voelcker. Toward a comprehensive formulation of nc verification as a mathematical and computational problem. *Journal of Design and Manufacturing*, 3:263–277, 1993.

[43] J.R. Miller. Sculptured surfaces in solid models: Issues and alternative approaches. *IEEE - Computer Graphics and Applications*, 6:37–48, 1986.

[44] K. Okushima and 1960 Hitomi. Plastic flow of metal in orthogonal cutting. *Transaction of Japan Society of Mechanical Engineering*, 26:423, 1960.

[45] H. Persson. NC machining of arbitrarily shaped pockets. *Computer-Aided Design and Manufacture*, 10(3):169–175, May 1978.

[46] L. Piegl. On nurbs: A survey. *IEEE - Computer Graphics and Applications*, 10(1):55–71, 1991.

[47] L. Piegl and W. Tiller. *The NURBS Book.* Springer-Verlag, 1995.

[48] P. Rigby. High speed milling in the mold and die making industries. In *Proceedings GE Superabrasives, Paris, France*, pages 2–14, 1993.

[49] P Siltanen and C. Woodward. Normal orientation methods for 3d offset curves, sweep surfaces and skinning. In *Proceedings Eurographics*, volume 10/3, pages C:449–457, 1992.

[50] S. Smith and J. Tlusty. An overview of modeling and simulation of the milling process. *ASME Journal of Engineering for Industry*, 113:169–175, 1991.

[51] Spatial Technology, Inc., Boulder, Colorado. *ACIS Geometric Modeler Application Guide*, 1994.

[52] Spatial Technology, Inc., Boulder, Colorado. *ACIS Geometric Modeler V3.0*, 1997.

[53] A.D. Spence. *Solid Modeller Based Milling Process Simulation.* PhD thesis, The University of British Columbia, Vancouver, BC, March 1992.

[54] A.D. Spence and Y. Altintas. CAD assisted adaptive control for milling. *ASME Journal of Dynamic Systems, Measurement and Control*, 113(3):444–450, September 1991.

[55] A.D. Spence, Y. Altintas, and D.G. Kirkpatrick. Integrating CAD with machining process control. In *CSME Mechanical Engineering Forum 1990–Symposium on Advanced Manufacturing Systems*, pages 53–58, Toronto, ON, June 1990. CSME. Volume III.

[56] Structural Dynamics Research Corporation. *I-DEAS GNC*, 1996.

[57] S. Takata. Generation of a machining scenario and its applications to intelligent machining operations. *CIRP Annals*, 42/1:531–534, 1993.

[58] S. Takata, M.D. Tsai, and T. Sata. A cutting simulation system for machinability evaluation using a workpiece model. *CIRP Annals*, 38/1:417–420, 1989.

[59] J. Tlusty and F. Ismail. Dynamic structural identification tasks and methods. *CIRP Annals*, 29(1):251–255, 1980.

[60] J. Tlusty and F. Ismail. Basic non-linearity in machining chatter. *CIRP Annals*, 30(1):299–304, 1981.

[61] J. Tlusty and F. Ismail. Special aspects of chatter in milling. *ASME Journal of Vibration, Acoustics, Stress, and Reliability in Design*, 105:24–32, 1983.

[62] J. Tlusty and P. MacNeil. Dynamics of cutting forces in end milling. *CIRP Annals*, 24:21–25, 1975.

[63] J. Tlusty and T. Mirowaki. Experimental and computational identification of dynamic structural models. *CIRP Annals*, 25(2):497–503, 1976.

[64] J. Tlusty, S. Smith, and C. Zamudio. New NC routines for quality in milling. *CIRP Annals*, 39(1):517–521, 1990.

[65] M.D. Tsai, S. Takata, M. Inui, F. Kimura, and T. Sata. Prediction of chatter vibration by means of a model based cutting simulation system. *CIRP Annals*, 39(1):447–450, 1990.

[66] M.D. Tsai, S. Takata, M. Inui, F. Kimura, and T. Sata. Operation planning based on cutting process models. *CIRP Annals*, 40(1):95–98, 1991.

[67] G.W. Vickers, H. Li, and Z. Dong. Automated rough machining of curved surface. *Proceedings of the CSME Forum SCGM*, pages 593–597, 1992.

[68] G.W. Vickers and K.W. Quan. Ball-mills versus end-mills for curved surface machining. *ASME Journal of Engineering for Industry*, 111(1):22–26, 1989.

[69] W.P. Wang. *Solid Geometric Modeling for Mold Design and Manufacturing.* PhD thesis, Cornell University, 1984.

[70] W.P. Wang. Application of solid modeling to automate machining parameters for complex parts. *Journal of Manufacturing Systems*, 17(1):57–63, 1988.

[71] W.P. Wang. Solid modeling for optimizing metal removal of three-dimensional NC end milling. *Journal of Manufacturing Systems*, 7(1):57–65, 1988.

[72] W.P. Wang and K.K. Wang. Geometric modeling for swept volume of moving solids. *IEEE - Computer Graphics and Applications*, 6:8–17, 1986.

[73] M. Weck, Y. Altintaş, and C. Beer. Cad assisted chatter-free nc tool path generation in milling. *International Journal of Machine Tool Design and Research*, 34(6):879–891, 1994.

[74] M. Yang and H. Park. The prediction of cutting forces in ball-end milling. *International Journal of Machine Tool Design and Research*, 31(2):45–54, 1991.

[75] I. Yellowley. Observations of the mean values of forces, torque, and specific power

in peripheral milling process. *International Journal of Machine Tool Design and Research*, 25(4):337–346, 1985.

[76] W.S. Yoo and B.K. Choi. Capp for die cavity machining. *Human Aspects In Computer Integrated Manufacturing*, pages 437–447, 1992.

[77] G. Yücesan and Y. Altintaş. Improved modelling of cutting force coefficient in peripheral milling. *International Journal of Machine Tool Design and Research*, 34(4):473–487, 1994.

[78] G. Yücesan and Y. Altintaş. Prediction of ball end milling forces. *ASME Journal of Engineering for Industry*, 118:95–103, 1996.

[79] I. Zeid. *CAD/CAM Theory and Practice.* McGraw-Hill, Inc., 1991.