GENERIC MECHANISTIC MODELING FOR MULTI-AXIS MACHINING

By

TREVOR E. BAILEY, M.Eng. (Mechanical Engineering)
McMaster University
Hamilton,Ontario, Canada

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Doctor of Philosophy

McMaster University

Generic Mechanistic Modeling for Multi-Axis Machining

DOCTOR OF PHILOSOPHY (2001)	McMaster University
(Mechanical Engineering)	Hamilton, Ontario

TITLE: Generic Mechanistic Modeling for Multi-Axis Machining

AUTHOR: Trevor E. Bailey
M.Eng. (Mechanical Engineering)
McMaster University
Hamilton,Ontario, Canada

SUPERVISOR: Dr. M.A.Elbestawi
Department of Mechanical Engineering
McMaster University

NUMBER OF PAGES: 1, 260

Abstract

A generic mechanistic approach for simulating multi axis machining of complex sculptured surfaces is presented. A generalized approach is developed for representing an arbitrary cutting edge design, and the local surface topology of a complex sculptured surface. A NURBS curve is used to represent the cutting edge profile. This approach offers the advantages of representing an arbitrary cutting edge design in a generic way, as well as providing standardized techniques for manipulating the location and orientation of the cutting edge. The local surface topology of the part is defined as those surfaces generated by previous tool paths in the vicinity of the current tool position. The local surface topology of the part is represented without using a computationally expensive CAD system. A systematic prediction technique is then developed to determine the instantaneous tool/part interaction during machining. The methodology employed here determines cutting edge in-cut segments by determining the intersection between the NURBS curve representation of the cutting edge and the defined local surface topology. These in-cut segments are then utilized as integration limits for a comprehensive force modeling methodology. A systematic model calibration procedure that incorporates the effects of varying cutting edge geometry, cutting speeds, and feed rates is developed. Experimental results are presented for the calibration procedure. Model verification tests were conducted with these cutting force coefficients. These tests demonstrate that the predicted forces are within 5% of experimentally measured forces.

An enhanced approach for dynamic mechanistic modeling for multi-axis machining is developed. The dynamic process simulation methodology is presented as a continuous solution for complex sculptured surface machining. The methodology is formulated to include the instantaneous deflection of the tool, and the tool deflection history over the entire tool path. The continuous dynamic process simulation methodology is compared with both static and dynamic simulation methodologies based on control points. The comparison is for linear cuts with constant and variable tool/workpiece immersion. This comparison shows that the modeling methodologies based on control points only offer a piece-wise continous representation of the static and dynamic cutting force and tool deflection over a given tool path. The accuracy of the cutting force prediction and the calculation of tool deflection using methodologies based on control points degrades for variable tool/workpiece immersion. The continous dynamic simulation methodology demonstrates a good representation of the static and dynamic cutting force and tool deflection for variable tool/workpiece immersion. The continous dynamic process simulation methodology is demonstrated for a complex sculptured surface machining operation. The rough machining operation for an airfoil-like surface is presented. The simulation results demonstrate how the continuous dynamic process simulation methodology is capable of predicting the cutting force and tool deflection for variable tool/workpiece immersions that occur during complex sculptured surface machining operations.

The generic simulation approach for multi-axis machining has been demonstrated as a process optimization tool. Feed scheduling was used to demonstrate the process optimization for multi-axis machining. A feed scheduling methodology for multi-axis machining was developed. The feed scheduling methodology was formulated based on maximum chipload and maximum force constraints. A case study for process optimization of machining an airfoil-like surface was used for demonstration. Based on the predicted instantaneous chip load and/or a specified force constraint, feed rate scheduling was utilized to increase metal removal rate. The feed rate scheduling implementation results in a 30% reduction in machining time for the airfoil-like surface without any sacrifice in the surface quality or part geometry.

The machine tool feed drive performance capabilities were integrated with the feed scheduling methodology. Two strategies were developed. The first strategy guaranteed that the scheduled feed rates were attainable within the machine Acc/Dec time constant, τ_{total}. The second strategy optimized the use of the feed drive capabilities while tracking the changes in cutting geometry along the tool path. A comparative study was performed for feed scheduling based on control points, Acc/Dec time constant, and optimized strategy. Both simulated and experimental results were given for each feed scheduling strategy. The criteria for comparison were the production savings, the machining integrity, and the machining safety. The optimized feed scheduling strategy was the unique strategy that improves simultaneously the production saving while guaranteeing the machining integrity.

The concept and implementation of an Internet-based facility for multi-axis milling process simulation and optimization was presented. The facility integrates the generic simulation approach for multi-axis machining, and state of the art multimedia and Internet technologies. The Internet-based facility for multi-axis milling process simulation and optimization provides a formalized system for designing and proofing

machining processes. Process develoment can occur iteratively, without the need for lengthy and/or expensive process trials. Implementing these development tools using internet technologies has several advantages. It allows all of the development tools to be in a centralized location, allowing for quick and reliable upgrade paths. Internet accessibility also gives quick and easy access to these development tools to users around the world.

Acknowledgements

I would like to express a sincere gratitude to my research supervisor, Dr. M.A. Elbestawi for his continuing guidance and support throughout this research. He has been a constant source of motivation and support.

I would also like to extend a sincere thank-you to all my colleagues at the Intelligent Machines and Manufacturing Research Centre who have helped create a work environment that is both supportive and inspiring.

I am also grateful for the financial support of my research through both NSERC and MMO.

I am especially thankful for my family's unfailing encouragement and support during my many years as a student.

Contents

List of Figures

List of Tables

Nomenclature

v	Cutting speed
f_r	Feed rate
N_{rpm}	Spindle RPM
r_c	Tool radius
s_t	Nominal feed per tooth
t_c	Instantaneous chip thickness
θ	Cutting rotation angle
P	Average cutting power
P_{sp}	Specific power
MRR	Metal Removal Rate
a	Axial depth of cut
d	Radial width of cut
N_{th}	Number of teeth on tool
$\Im_c$	Average cutting force
$\Im_s$	Thrust cutting force
$\Im_t$	Tangential cutting force
$\Im_r$	Radial cutting force
K_t	Tangential cutting pressure constant
K_r	Radial cutting pressure constant
DZ_0	Current normal deflection of tool
DZ_1	Normal deflection of tool (1 tooth period)
DZ_2	Normal deflection of tool (2 tooth period)
DZ_3	Normal deflection of tool (3 tooth period)
m	Equivalent modal mass
k	Modal stiffness
c	Modal damping coefficient

x	Displacement
$\dot{x}$	Velocity
$\ddot{x}$	Acceleration
$\Im_n$	Normal cutting force
$\Im_f$	Friction cutting force
M_t	Constants characterizing the size effects of the workpiece
M_r	Constants characterizing the size effects of the workpiece
P_i	NURBS curve control points
ω_i	NURBS curve weights
$N_{i,p}(u)$	NURBS pth B-spline basis functions
U	NURBS non-periodic knot vector
u_i	NURBS parameter
$Z_{surf,i}$	Swept surface Z value
Z_{min}	Minimum of all swept surface Z values
P_i	Point on cutting edge
$\vec{\tau}$	Unit vector tangent to cutting edge
β_z	Instantaneous cutter helix angle
$\vec{Z}$	Unit vector along tool z-axis
φ	Lag angle
θ_i	Instantaneous angle of rotation
θ_0	Reference angle of rotation
ρ_i	Tool lead angle
R_0	Nominal tool radius
α_i	Instantaneous rake angle
α_0	Nominal rake angle
κ	Vector on rake face
$\vec{V}_u$	Vector used to define κ
$\vec{n}_r$	Vector normal to rake face
$\vec{b}$	Vector perpendicular to $\vec{\tau}$ in plane defined by $\vec{\tau}$ and κ
$\vec{n}_{fr}$	Vector on rake face in chip flow direction
η_c	Chip flow angle
ξ	Vector on clearance face
$\vec{T}_u$	Vector used to define ξ
γ_i	Instantaneous clearance angle
$\vec{n}_e$	Vector normal to clearance face
$\vec{n}_{fe}$	Vector perpendicular to $\vec{\tau}$ in plane defined by $\vec{\tau}$ and $\vec{n}_e$

δS	Length of an infinitesimal cutting edge segment
λ	Current tool path inclination angle
δA_r	Differential contact area
$\delta \Im_{nr}$	Differential cutting force normal to rake face
$\delta \Im_{fr}$	Differential cutting force on rake face in friction direction
$\delta \Im_{ne}$	Differential cutting force normal to clearance face
$\delta \Im_{fe}$	Differential cutting force on clearanc face in friction direction
K_{nr}	Pressure cutting coefficient on rake face
K_{fr}	Friction cutting coefficient on rake face
K_{ne}	Pressure cutting coefficient on clearance face
K_{fe}	Friction cutting coefficient on clearance face
$\vec{\Im}_p$	Total cutting force (x,y,z)
$\vec{\Im}_{nr}$	Total cutting force normal to rake face
$\vec{\Im}_{fr}$	Total cutting force on rake face in friction direction
$\vec{\Im}_{ne}$	Total cutting force normal to clearance face
$\vec{\Im}_{fe}$	Total cutting force on clearance face in friction direction
K_i	Instantaneous cutting coefficient (i=1,2,3,4,5)
A_i	Geometric coefficient (i=1,2,3,4,5)
B_i	Geometric coefficient (i=1,2,3,4,5)
C_i	Geometric coefficient (i=1,2,3,4,5)
V_j	Instantaneous cutting speed
β_j	Geometric constant
a_j	Feed per tooth coefficients (j=1,2,3)
K_r	Rake face coefficient matrix
K_f	Clearance face coefficient matrix
$\hat{R}$	Rake face vector matrix
$\hat{E}$	Clearance face vector matrix
$\Im_m$	Experimental cutting force (x,y,z)
$s_{t,i}$	Effective feed per tooth (with runout)
r_i	Cutting edge radial runout
$ICSeg[i][j]$	Cutting edge incut segment
Ax_{inc}	Number of axial elements
dz	Minimum axial distance
N_{rot}	Number of user specified tool rotations
n_{step}	Number of simulation time steps for each tool path segment
TP_{len}	Tool path segment length

f_{rev}	Feed per revolution
N_{rev}	Number of simulation steps per revolution
T_s	Translation in direction of tool path tangent vector
T_d	Translation normal to tool path
$P(k)$	Feed scheduling control points
P_0	Initial position of the tool
P_e	End position of the tool
P_T	Target position at end of tool path
ΔX	Feed scheduling control point spacing
F_S	Commanded feed rate
F_{max}	Maximum allowable feed rate
F_{min}	Minimum allowable feed rate
$\Im$	Cutting force
$\Im_{lim}$	Maximum allowable cutting force
$\delta\Im$	Cutting force tolerance
F_a	Actual machine feed rate
τ_{total}	Machine Acc/Dec time constant
τ_{acc}	Machine Acc time constant
τ_{dec}	Machine Dec time constant
T	Tooth passing period
N_c	Number of FIR filter coefficients
I_b	Feed scheduling block
C_j	FIR filter coefficients
F_{S0}	Intermediate commanded feed rate
δP	Remaining path distance
N_a	Number of added time increments
N	Number of time increments (Optimized Strategy)
F_{aT}	Target actual machine feed rate
I	Reference index (Optimized Strategy)
F_{ST}	Target commanded feed rate
F_a^*	Unknown actual machine feed rate
F_S^*	Unknown commanded feed rate
T_M	Time for a position move
$u(t)$	System input (Identification)
$y(t)$	System output (Identification)
G_p	Plant transfer function

H	Noise transfer function
z^{-1}	One step delay operator
A	Model polynomial (Identification)
B	Model polynomial (Identification)
C	Model polynomial (Identification)
D	Model polynomial (Identification)
F	Model polynomial (Identification)
n_k	Number of sample delays
n_a	Order of A polynomial(Identification)
$n_b - 1$	Order of B polynomial(Identification)
n_c	Order of C polynomial(Identification)
n_d	Order of D polynomial(Identification)
n_f	Order of F polynomial(Identification)
$\hat{\sigma}_{ue}$	Estimated cross-correlation between input and residuals
σ_{ue}	Tolerance on the standard deviation of $\hat{\sigma}_{ue}$
$\hat{\bar{\sigma}}_e$	Estimated value on the whiteness of residuals
σ_e	Tolerance value on the whiteness of residuals
$\hat{\sigma}_a$	Estimated value on the deviation of FRF amplitude
σ_a	Tolerance value on the deviation of FRF amplitude
$\hat{\sigma}_p$	Estimated value on the deviation of FRF phase
σ_p	Tolerance value on the deviation of FRF phase
$\hat{\sigma}_f$	Estimated value on the simulation fit
σ_f	Tolerance value on the simulation fit
$\hat{\sigma}_{ZP}$	Estimated plant pole
$\hat{\sigma}_{P+}$	Real part of pole
$\hat{\sigma}_{ZZL}$	Estimated value on the deviation of zeros
σ_{ZZL}	Tolerance value on the deviation of zeros
$\hat{\sigma}_{ZPL}$	Estimated value on the deviation of poles
σ_{ZPL}	Tolerance value on the deviation of poles
ΔF	Commanded feed rate step

Chapter 1

Introduction

Complex scuptured surfaces are common features in parts and/or components that are produced in the die and mold, automotive, aerospace, and appliances industries. The representation of these surfaces may be approximated by a collection of plane-faced polyhedral with straight edges that involve hundreds of faces and edges or by a collection of curved surface patches bounded by curved edges. These surfaces have been mathematically defined in various ways; including Bezier, B-spline, and Non-uniform rational B-spline (NURBS) surfaces [33, 69]. Machining these surfaces using a computer numerical control (CNC) machine has been a challenging task due to problems associated with interpolating points on the complex surface. Commercially available CAD/CAM systems use different methodologies for design and manufacture of complex sculptured surfaces. The main objective for these systems has been to generate cutter path motions that increase the efficiency of the machining process, eliminating surface gouging, and reducing the surface roughness (cusp height). Process planning and optimization has been restricted to geometric modeling and tool path planning. Optimizing the process parameters (feed per tooth, cutting speeds, cut geometry) has not typically been addressed by commercially available CAD/CAM systems. The process parameters are usually selected from available databases in machining handbooks. The values given by the machining handbooks are primarily intended for machining parts with polyhedral features such as slots, holes, pockets, and faces where the cut geometry is held relatively constant. Incorrectly determining the process parameters for complex sculptured surface machining can result in inferior products, tool breakage, and lower productivity.

The optimization of the process parameters should be based on the physics of the interaction between the tool and workpiece, the values of the cutting forces, tool/part deflections, and temperatures generated during machining. During machining of complex sculptured surfaces the process parameters (cut geometry and feed per tooth) vary. These instantaneous changes to the process parameters results in variable cutting forces, temperatures, and stresses acting on the tool. The effect is to degrade tool performance and the surface quality of the part. In addition, the selected cutting speed can cause chatter or forced vibration that will chip the tool and/or reduce the quality of the produced surface.

It is clear from the above discussion that a model-based machining simulation is required to allow process simulation and optimization for complex sculptured surface machining. Spence [79] used a constructive solid geometry (CSG) part representation to perform flat-end milling simulations. The instantaneous immersion geometry (entry and exit angles) was determined by sweeping a semi-circle along the tool path. El-Mounayri *et al* [28] simulated ball-end milling using a B-rep polyhedral-based solid modeler. The tool cutting edge was modeled with a cubic Bezier curve. The removed volume was determined using the B-rep solid modeler. The in-cut segments were determined by intersecting the removed volume with the cutting edge. Imani [39] developed a model-based simulation methodology for die cavity milling operations. The simulation integrates a geometric modeling environment with a mechanistic force model. A commercially available B-rep geometric modeler (ACIS) was used. The immersion geometry is computed using the contact face between the ball-end mill and the machined part. Abrari [3] developed a model-base simulation methodology similar to Imani. Abrari used the same geometric modeler (ACIS) for development. Abrari's research focus was multi-axis milling simulation of flexible parts. A dynamic force model was developed for modeling the tool/workpiece structural interaction. The dynamic response of the workpiece to instantaneous forces was modeled using a finite element approach. A solid model based automatic mesh generation algorithm was also developed.

All of the model-based machining simulations above incorporate a solid modeler. The efficient implemention of a solid modeling approach for sculptured surface machining requires a powerful spatial tool kit and/or a geometric engine. The commercially available spatial tool kits and geometric engines that can adequately simulate both complex tool and workpiece geometries are cost prohibitive to many industries. In addition, only global surface topology information is utilized to represent the workpiece using commercial systems. The global surface topology representations reduce the accuracy of the machining simulation. An enhancement for this type of modeling is to use local surface topology in the analysis. The ability to model the local surface topology of the workpiece is necessary to accurately determine the tool/workpiece interaction during machining. Modeling the local surface topology of the workpiece accounts for the material removed by previous tool paths in the vicinity of the current tool location.

The key factors necessary for accurately modeling the machining of complex sculptured surface parts are:

- A generalized representation of an arbitrary cutting edge design.
- A generalized representation of the local surface topology of a complex sculptured surface part.
- A systematic prediction technique for determining the instantaneous tool/workpiece interaction during machining.
- Allowing the use of an arbitrary tool path specification (CLData file)
- Providing the ability to integrate multiple machining operations within the simulation (ie. specified by multiple tools and CLData files).

1.1 Research Objectives

There are three main objectives for the proposed research. The first objective is to develop a generic simulation approach for multi-axis machining. To meet this objective the following tasks are identified:

- Develop a geometric modeling scheme that is generic to multi-axis machining. The modeling should offer a generalized approach to representation of the cutting edge, the local surface topology of the part, and the tool/part interaction.
- Develop a comprehensive mechanistic force modeling methodology for multi-axis machining.

- Develop a systematic model calibration procedure that incorporates the effects of varying cutting edge geometry, cutting speeds, and feed rates.
- Develop a continuous dynamic force simulation methodology for multi-axis machining.

A second objective is to demonstrate how the generic simulation approach for multi-axis machining can be used as a process optimization tool. Feed scheduling is used to maximize feed rate subject to specific process constraints for multi-axis machining. The following tasks are identified to meet this objective:

- Develop a feed scheduling methodology for multi-axis machining.
- Incorporating machine tool feed drive performance capabilities with the feed scheduling methodology.

The third objective is to integrate the generic simulation approach for multi-axis machining with internet technologies to provide an Internet enabled process simulation and optimization environment.

1.2 Thesis Outline

The thesis will be presented as follows:

In chapter 2 a literature review is presented. The scope of this thesis includes research areas related to process modeling, process optimization and internet technologies. This chapter provides a detailed overview of these research areas as they relate to the research objectives given in section 1.1.

Chapter 3 presents the modeling methodologies for a generic simulation approach for multi-axis machining. A geometric modeling scheme offering a generalized approach for representing the tool cutting edge, the local surface topology of the part, and the tool/part interaction during machining is presented. A comprehensive force modeling methodology for multi-axis machining is also given in this chapter. A systematic model calibration procedure that incorporates the effects of varying cutting edge geometry, cutting speeds, and feed rates is developed. A dynamic process simulation methodology is also presented. The concept and implementation of an Internet based facility for multi-axis milling process simulation and optimization is also described in this chapter.

The feed scheduling methodologies are developed in chapter 4. A feed scheduling methodology for multi-axis machining is presented. This chapter also presents a methodology for integrating the machine tool feed drive performance capabilities with feed scheduling.

Simulated and experimental results are presented in chapters 5 and 6. Chapter 5 provides an overview of the simulations and experimental verification for the modeling methodologies presented in chapter 3. This chapter also discusses various applications of the Internet based machining system. Chapter 6 demonstrates the feed scheduling concept for the methodologies presented in chapter 4.

The major contributions of this thesis are summarized in chapter 7. Recommendations for future extensions to this work are also given in this chapter.

Chapter 2

Literature Review

2.1 Introduction

The scope of this thesis includes research areas related to process modeling, process optimization and Internet technologies for machining systems. This chapter provides a detailed overview of these research areas as they relate to the research objectives given in section 1.1. A review of process modeling is given in section 2.2. This review discusses both flat and ball milling models as well as model calibration techniques. Section 2.3 discusses research for modeling complex sculptured surface machining. Feed scheduling is the chosen process optimization method for this thesis. Feed scheduling is discussed in section 2.4. Internet based machining systems are reviewed in section 2.5.

2.2 Process Modeling

Process modeling has been extensively studied in the literature. Both mechanistic and semi-empirical cutting force models have been presented. In metal cutting research, mechanistic cutting force models are based on the general mechanisms of chip formation and tool flank - workpiece contact. The cutting forces are determined from pressure and friction coefficients on the rake and flank contact surfaces. For milling these models have been classified primarily based on the cutter geometry. The two main classifications are flat and ball end milling.

2.2.1 Flat End Milling Models

To adequately define any cutting process the cutting speed, v, and the feed rate, f_r must be specified. For milling this translates into defining a spindle speed, $N_{rpm} = \frac{v}{r_c}$ (r_c: cutter radius), and a feed per tooth, s_t. As the part is moved toward the cutter the path traced by a single tooth is trochoidal [62]. However, because the rotational speed is usually much higher than the feed per tooth, a circular path can be assumed. Accordingly, the instantaneous chip thickness varies sinusoidally as the cutter rotates. Figure 2.1 shows that the instantaneous chip thickness, t_c, can be determined at any immersion angle θ by:

$$t_c = s_t \sin\theta \tag{2.1}$$

Models based on equation 2.1 have been extensively published in the literature. An overview of several models for the milling process was provided by Smith and

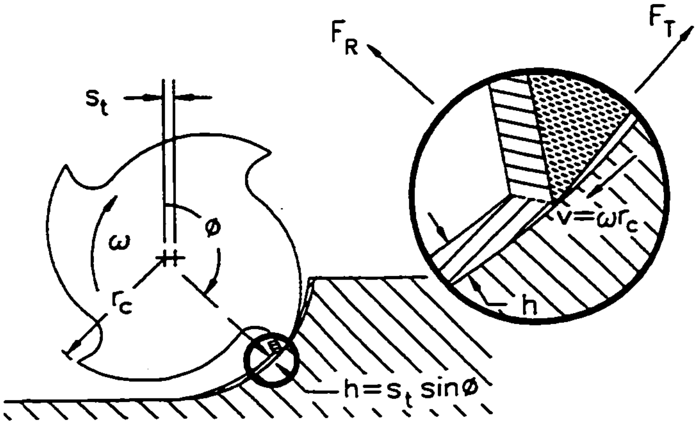


Figure 2.1: Sinusoidally varying chip thickness

Tlusty [76]. This paper classified and evaluated the various models based on the methods of force computation and deflection feedback to the force. The models were presented in ascending order of sophistication and accuracy.

Average Rigid Force, Static Deflection Model This is the most basic model. It is based on the relationship between the metal removal rate (MRR) and the average power consumed during cutting, P. The governing relation is:

$$P = P_{sp}MRR \tag{2.2}$$

The parameter P_{sp}, termed specific power, has been experimentally determined for common materials and is available in the literature [64]. The MRR is determined from the incut parameters

$$MRR = adN_{th}s_tN_{rpm} \tag{2.3}$$

where

a axial depth of cut

d radial depth of cut

N_{th} number of teeth on the cutter

s_t chip load (feed per tooth)

N_{rpm} spindle speed

The average cutting force, $\Im_c$, is approximated as

$$\Im_c = P_{sp}MRR/v \tag{2.4}$$

where v is the peripheral cutting speed. As a first approximation the thrust force, $\Im_s$, acting on the cutter to cause deflection normal to the cut is given as

$$\Im_s = \frac{1}{2}\Im_c \tag{2.5}$$

The deflection of the cutter is then computed by assuming that the thrust force is concentrated at the tip of the cutter and that the cutter can be modelled as a cantilever beam. This deflection is then taken as the surface error for the milled surface.

This model calculates the cutting force based on an average cutting power. Hence, the cutting force is given as an average value. Also, the deflection of the cutter is not used in the force computation (the cutter is assumed rigid). The deflection of the cutter is taken as simply being proportional to the average force. Inertia effects of the tool, spindle, and workpiece have been neglected in force calculation. This model has been widely used as a 'first approximation', to predict average cutting power and torque [95]. However, it lacks the necessary sophistication to adequately predict the instantaneous cutting forces and deflections. The instantaneous variation of the cutting force and deflection as well as their histories must be considered.

Instantaneous Rigid Force, Static Deflection Models This class of models considers an instantaneous force acting on the cutter. The force is computed as a vectoral sum of the incremental forces along each helical cutting edge for each tooth in cut, figure 2.2 [76]. Again the deflection of the cutter is not considered in the force formulation.

The force calculations were based on Sabberwal's observation that the tangential force acting on an incremental tooth element was proportional to the frontal area of the chip being removed and that the radial component of the force is proportional to the tangential force [74]. The tangential and radial forces are given as:

$$\Im_t = K_t a t_c \tag{2.6}$$

$$\Im_r = K_r \Im_t \tag{2.7}$$

where

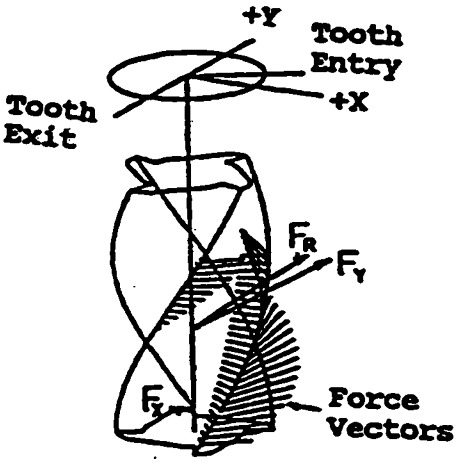


Figure 2.2: Instantaneous Rigid Force Model [76]

K_t	constant related to the specific tool/workpiece material, cuttings speeds/feeds and cutting geometry
K_r	constant related to the specific tool/workpiece material, cuttings speeds/feeds and cutting geometry
a	axial depth of cut
t_c	instantaneous chip thickness

The cutter deflection is static and taken as simply proportional to the force.

Although this model calculates the instantaneous cutting forces, it does not consider the effects of deflection histories. There must be feedback incorporated into the force model to account for both the current and previous deflections.

Instantaneous Force with Static Deflection Feedback This model uses the static deflection as feedback in the calculation of the force [81]. It has been assumed that the deflection of the cutter only affects the feed per tooth. Incorporating feedback, equation 2.6 becomes:

$$\Im = K_t a(s_t - \Im/k) \tag{2.8}$$

where k is the stiffness of the cutter. Solving for the force

$$\Im = [K_t a/(1 + K_t a/k)]s_t \tag{2.9}$$

As the cutter deflects, the thickness of the formed chip is less than the nominal feed per tooth. The next tooth will see a higher chip load, acting to increase the cutting force. This demonstrates that the deflection histories (surfaces generated by previous teeth) plays an important role in the calculation of the cutting forces.

Regenerative Force, Dynamic Deflection Model The regenerative force, dynamic deflection model incorporates the effects of the feed per tooth, the current deflection of the cutter, and the deflection histories in the calculation of the force [40]. The model is termed regenerative because it considers the surface that is produced by previous teeth. The force on any tooth is given as:

$$\Im_{i+1} = K_t a(t_c + DZ_0 - min(DZ_1, DZ_2, DZ_3)) \tag{2.10}$$

where

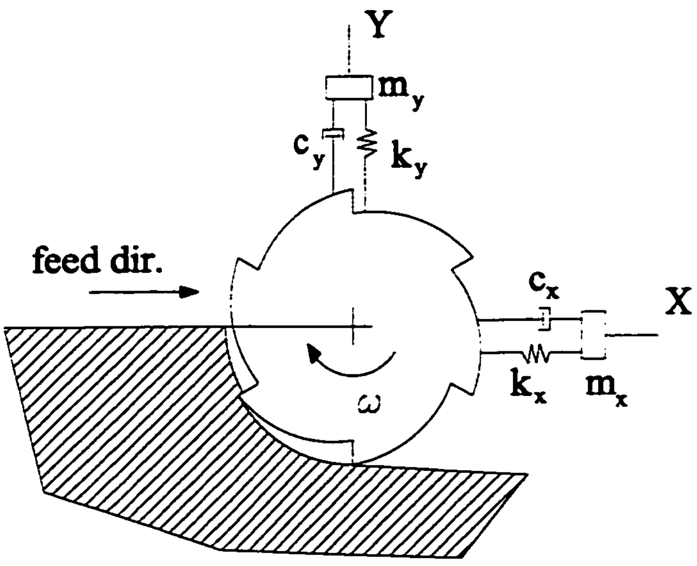


Figure 2.3: Two-Degree-of-Freedom Model of Milling Cutting [76]

i time step counter

DZ_0 current deflection of the cutter normal to the surface of the cut

DZ_1, DZ_2, DZ_3 are the deflections of the previous 3 tooth periods

This model also accounts for the system inertias (tool, workpiece, spindle). Figure 2.3 [76] depicts how the dynamics of the cutter have been modeled. The model considers a two degree of freedom system in two mutually perpendicular directions. The modal parameters; equivalent mass, damping and stiffness are typically determined experimentally using impact tests.

During the simulation, the instantaneous force on each tooth is computed, then vectorally summed to give the total force at each angular position of the tool. From the force, the coordinate accelerations can be computed. By integrating the acceleration twice the displacement of the tool can be found. For each coordinate the governing equation for force is:

$$\Im_i = m\ddot{x}_i + c\dot{x}_i + k_i \tag{2.11}$$

Integration can be approximated using Newton's method

$$\begin{align}
\ddot{x_{i+1}} &= (\Im_i - c\dot{x}_i - kx_i)/m \tag{2.12}\\
\dot{x_{i+1}} &= \dot{x}_i + \ddot{x}_i\Delta t \tag{2.13}\\
x_{i+1} &= x_i + \dot{x}_i\Delta t \tag{2.14}
\end{align}$$

where m is the equivalent mass, k is the stiffness, c is the damping coefficient, x is the displacement, $\dot{x}$ is the velocity, and $\ddot{x}$ is the acceleration.

The deflection of the cutter normal to the cut is given by

$$z_i = x_i \sin\theta + y_i \cos\theta \tag{2.15}$$

As the simulation progresses, the position of any tooth in contact with the workpiece is stored as a record of the surface left behind. In this way, the chip thickness can be adjusted for both the current deflection of the cutter and the surface left by the passage of previous teeth. The model includes nonlinearities that occur in the physical machining process. If the vibration amplitude is large enough, the tooth will loose contact with the workpiece. Equation 2.10 produces a negative force when this occurs. A negative force is meaningless and a condition must be included to account for this phenomena.

$$if\ \Im_i \;<\; 0 \quad \Im_i = 0 \tag{2.16}$$

The addition of regenerative feedback into the force model allows it to correctly simulate chatter and forced vibration phenomena for milling operations. However, the model is sensitive to the chosen time step for iteration during simulation. Large time steps result in a decrease in accuracy, excessively small time steps may cause the integration method to become unstable.

Closed Form Models All the models presented above have obtained the force along a helical cutting edge by summing the instantaneous forces at incremental locations along the helix. The class of models presented in this section attempt to analytically perform the integration of the force along the helical cutting edge. This form of model has historically been closely coupled with CAD simulations for the milling process. Analytic expressions have been reformulated into closed forms suitable for solid modeler, CAD based systems.

Spence *et al.* [78] developed a mechanistic force model for end milling with constant axial depth of cut. The final force formulation provides an instantaneous resultant force for each angular position of the cutter. It has the form:

$$\Im(\theta) = 0.5 s_t K_t (1 + K_r^2)^{\frac{1}{2}} \Im_{cad}(\theta) \tag{2.17}$$

where $\Im_{cad}$ is independent of the cutter calibration parameters K_t, K_r and is solely based on the analytic integration along the incut portion of the helical tooth. The

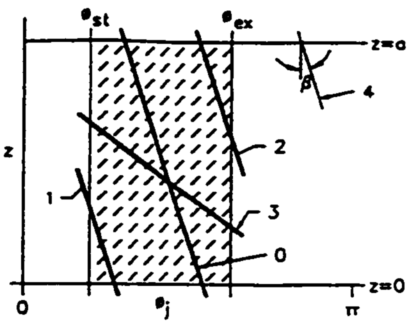


Figure 2.4: Intersection Zones and Cases [78]

axial integration limits for each helical flute are supplied by a CAD solid modeller. They were classified based on how the part face section interacted with the helical tooth (figure 2.4 [78]).

This formulation for the forces in milling has particular advantages for process planning purposes. When the model is coupled with a CAD solid modeler it can predict undesirable transients in the force to prevent catastrophic failures such as tooth and shank breakage. This is because the force is proportional to $\Im_{cad}$, a geometric quantity, which can be determined explicitly prior to actual machining. The computational efficiency of this model also allows it to be used for online identification, monitoring and control applications. This model is, however, limited to constant axial depth of cut (two dimensional) and can not be used for sculptured surface machining.

Other analytic expressions for the force in end milling have been carried out separately by Armarego *et al.* [7], Yucesan *et al.* [107], and Wang *et al.* [93]. Armarego's analytic expressions modeled the elemental components of the force as oblique. The components of force due to an elemental oblique cut were expressed as functions of the cutting and edge force intensities. These parameters were predicted based on orthogonal cutting tests. The integration limits for this model was the angular engagement of the cutter. This model was capable of predicting three dimensional forces (x,y,z). The analytic expressions were, however, not conveniently formulated. They did not provide an explicit geometric expression proportional to the elemental force which could be used for process planning.

Yucesan's mechanistic cutting force model was based on a force equilibrium on the cutter rake surface. This required that the cutter surface geometry be defined as a curvilinear coordinate sytem along the cutting edge. The integration of elemental forces acting on this surface produced the analytic expressions.

The elemental forces (δF_n, δF_f) can be integrated over the engagement length of the cutter to give the overall effects of normal and frictional forces

$$\Im_n = -\int n\, K_t\, \delta A_r \tag{2.18}$$

$$\Im_f = \int T_c\, K_t\, K_r\, \delta A_r \tag{2.19}$$

The integration limits are functions of the instantaneous angular position of the cutter and its tooth engagement angle.

This model provides analytic expression for the forces in three coordinate directions (x,y,z). After integration its final form is similar to Spence [78] in that the forces are proportional to a purely geometric quatity.

Wang *et al.* [93] presents a closed formulation for the dynamic forces in end milling. The dynamic forces are expressed as explicit functions of cutting parameters and tool/workpiece geometry. The cutting force is formulated as the angular domain convolution of three cutting process component functions. These functions include the elementary cutting function, the chip width density function, and the tooth sequence function. The elementary cutting force function describes the chip formation process for an elemental cutting area. The elementary cutting force function is characterized by the chip thickness variation and radial cutting configuration. The chip width density function defines the chip width per unit cutter rotation along a cutter flute within the range of axial depth of cut. The tooth sequence function is defined by the tooth pitch of the tool and order of tooth engagement. The force model is transformed into the frequency domain. This transformation allows the Fourier series coefficients of the cutting forces to be defined explicitly as algebraic functions of tool parameters and cutting conditions. This force model formulation presents a mechanism for investigating the effects of various process parameters on cutting force prediction.

In general, the above models for end milling are restricted to static cutting conditions. Process dynamics and system inertias are not considered in these formulations. They are, however, computationally effecient and can be integrated with CAD modeling software to provide a scheduling tool for milling operations.

2.2.2 Ball End Milling Models

Ball end milling is one of the most widely used cutting processes for machining sculptured surfaces in the die/mold and aerospace industries. The primary reason for their wide spread implementation is their ability to be easily positioned relative to curved surfaces. Often ball end mills only require two dimensional cutter compensations,

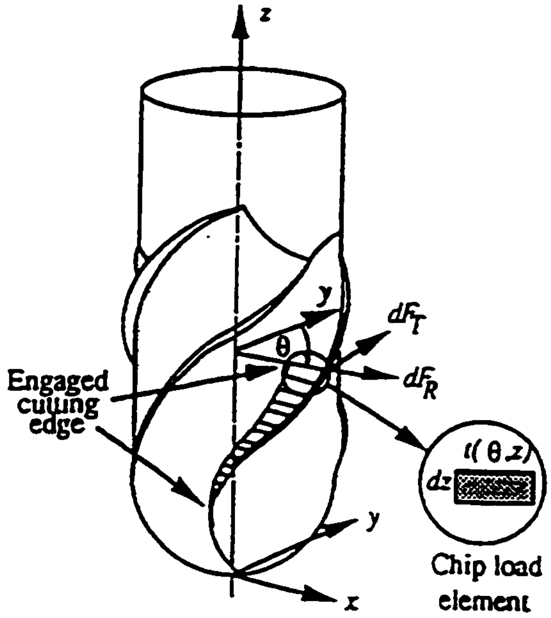


Figure 2.5: Ball end mill with Chip load element [2]

generating simpler NC machine programs.

The cutting action of ball end mills was first studied by Hosoi [37]. However, he made no attempt to study the mechanics of cutting with ball end mills, limiting his investigation to mechanisms for redesigning the cutting tool to avoid cutter breakage and edge chipping.

Investigation of the mechanics of ball end milling has only recently been seen in the literature. An experimentally verified model for ball end mills was recently published by Feng *et al.* [32]. This model clarified the misconceptions that the cutting mechanics of the ball portion of the tool could be expressed as a series of end mills with varying diameters [57]. Feng's model formulated the forces based on the chip geometry and the undeformed chip load distribution along the engaged cutting edges. The tangential and radial components of the cutting force, $\delta\Im_t$, and $\delta\Im_r$ at a differential cutting element (figure 2.5 [32]) is given by:

$$\delta\Im_t = K_t(Z)\,\delta Z[t_i(\theta, Z)]^{M_t} \tag{2.20}$$

$$\delta\Im_r = K_r(Z)\,\delta Z[t_i(\theta, Z)]^{M_r} \tag{2.21}$$

where δZ is the differential width of cut , $t_i(\theta, Z)$ is the undeformed radial chip thickness, M_t and M_r are constants characterizing the size effects of the workpiece, and $K_t(Z)$ and $K_r(Z)$ are the specific cutting forces (approximated by polynomials). To obtain the instantaneous cutting forces on the ball end mill, the elemental tangential and radial cutting forces of all engaged teeth are resolved into the reference coordinate system and summed. Feng's force model has been formulated assuming that the cutter is rigid. The system dynamics and inertias have not been considered. This model has a tendency to over estimate the cutting forces [2].

A mechanistic model for ball end milling was presented by Yucesan [105]. The model is based on an analytic representation of the ball shaped helical flute geometry, and its rake and clearance surfaces. The friction and pressure forces ($\Im_n$, $\Im_f$) on the rake face are assumed proportional to the uncut chip thickness area in a direction defined by θ_c (figure 2.6 [2]). The forces exerted on the flank contact face ($\Im_{nf}$, $\Im_{ff}$) are concentrated on the in cut portion of the cutting edge. The total cutting force is expressed as a function of tool rotation angle (θ) and immersion (α) of the flutes.

$$\begin{aligned}\vec{\Im}(\theta) &= \int_{\alpha_1(\theta)}^{\alpha_2(\theta)} K_n[-\vec{n}\,(\alpha+\theta) + K_f(\alpha,\theta)\,\vec{T}_c(\alpha+\theta)]\delta A_r \\ &\quad + \int_{\alpha_1(\theta)}^{\alpha_2(\theta)} K_{nf}[-\vec{n}_f\,(\alpha+\theta) + K_{ff}\,\vec{\tau}_f(\alpha+\theta)]\delta A_f\end{aligned} \tag{2.22}$$

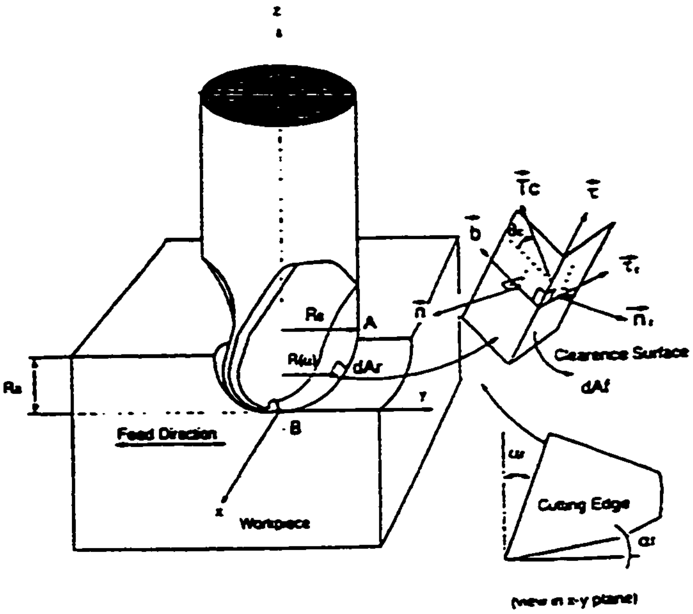


Figure 2.6: Geometry and Tool Coordinates for a Ball Mill Cutter [105]

where δA_r is the area of an infinitesimal rake surface element. The integration limits are functions of the cutter geometry, immersion of the cutter in the workpiece, feed direction and the tool rotation angle. This model is based on general mechanisms of chip formation and tool flank - workpiece contact. The cutting forces are determined through experimental determination of pressure and friction coefficients on the rake and flank contact surfaces. It is capable of predicting forces in three cartesian directions for unique axial and radial depths of cut, and feed rate.

A simpler model for the mechanics of ball end milling was provided by Jain and

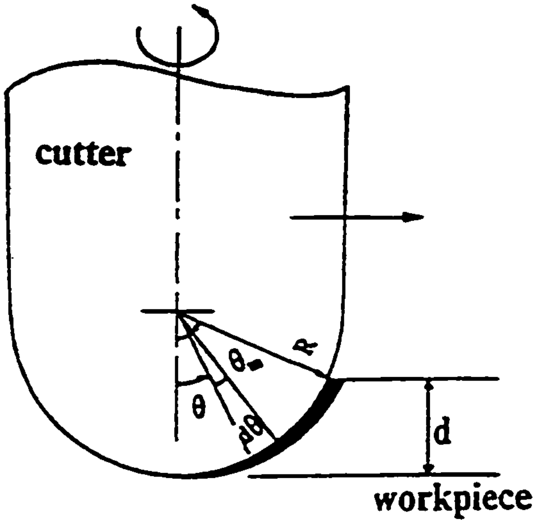


Figure 2.7: Chip Section Geometry in Ball End Milling [2]

Yang [41], though no experimental verification was presented. The normal and tangential components of the cutting force were given as:

$$\Im_t = K_t a s_t \sin\theta \tag{2.23}$$

$$\Im_r = \eta_r K_r a s_t \sin\theta \tag{2.24}$$

where K_t, K_r, and η_r are cutting constants, a is the depth of cut, and s_t is the feed per tooth (figure 2.7 [2]).

A model presented by Imani [39] for ball end milling is similar in structure to Yucesan's model [105]. A methodology is implemented which incorporated the effects of varying axial depth of cut when experimentally determining the pressure and friction coefficients.

Abrari [3] developed a closed form, analytic expression for the forces in ball end milling. The coordinate forces are assumed to be functions of the projected chip load areas. An average specific force matrix K was determined experimentally to include the effects of normal and shear forces. Both Imani [39] and Abrari [3] use a commercially available geometric modeler (ACIS) to represent the cutting edge, and calculate the tool/part intersection.

2.2.3 Model Calibration

The majority of the models presented in sections 2.2.1 and 2.2.2 require a set of calibration coefficients. Several techniques for determining these coefficients have been presented in the literature. Budak *et al*[16] presented a procedure for predicting milling force coefficients based on orthogonal cutting data. Ehmann *et al*[27] proposed a calibration procedure for determining constant cutting force coefficients. The cutting force coefficients were assumed constant irrespective of the cutting conditions and cutter rotational angle. A key feature of this methodology was the provision of a systematic procedure for synchronizing experimental cutting forces with those generated by a simulation. Force peaks were used for synchronization. Yucesan *et al*[105] assume an average value for the cutting force coefficients, fitting them to experimental data using a linear regression analysis. Imani [39] provided a calibration procedure specific to ball end milling. The cutting force coefficients were expressed as a function of axial depth of cut. Indirectly, this calibration procedure accounts for the variation in cutting speed for ball end milling. The calibration procedure uses a constrained optimization algorithm (Sequential Quadratic Programming) to fit experimental data. A calibration procedure proposed by Kapoor *et al*[43] expresses the pressure and friction cutting force coefficients as a function of chip thickness and cutting speed. This procedure is applied to insert end milling. Each coefficient is expressed as an exponentially decaying function of chip thickness and cutting speed.

2.3 Complex Sculptured Surface Machining

Complex scuptured surfaces are common features in parts and/or components that are produced in the die and mold, automotive, aerospace, and appliances industries. The representation of these surfaces may be approximated by a collection of plane-faced polyhedral with straight edges that involve hundreds of faces and edges or by a collection of curved surface patches bounded by curved edges. These surfaces have been mathematically defined in various ways; including Bezier, B-spline, and Non-uniform rational B-spline (NURBS) surfaces [33, 69]. Machining of such surfaces using a computer numerical control (CNC) machine has been a challenging task due to problems associated with interpolating points on the complex surface. Therefore, different methodologies have been developed for design and manufacture of complex sculptured surfaces. Several researchers have investigated the development and optimization of a machining strategy for such surfaces in the last decade [61, 12, 56, 36, 30, 35, 44, 60, 23, 55] . Their main concern has been to generate cutter path motions that increase the efficiency of the machining process, eliminating surface gouging, and reducing the surface roughness (cusp height). Very few systems appear to have been developed that allow for highly accurate machining of complex sculptured surfaces within user specified tolerances. Cutting tool geometry was included in most of these methodologies and angles of tool inclination were optimized to reduce the sculpt's height in [72]. The process planning and optimization discussed in the references above are restricted to geometric modeling and tool path planning. Optimizing the process parameters (feed per tooth, cutting speeds, cut geometry) was not investigated in the above research. The process parameters were usually selected from available databases in machining handbooks. The values given by the machining handbooks are primarily intended for machining parts with polyhedral features such as slots, holes, pockets, and faces where the cut geometry is held relatively constant. Incorrectly determining the process parameters for complex sculptured surface machining can result in inferior products, tool breakage, and lower productivity.

The optimization of the process parameters should be based on the physics of the interaction between the tool and workpiece, the values of the cutting forces, tool/part deflections, and temperatures generated during machining. During machining of complex sculptured surfaces the process parameters (cut geometry and feed per tooth) vary. These instantaneous changes to the process parameters results in variable cutting forces, temperatures, and stresses acting on the tool. The effect is to degrade tool performance and the surface quality of the part. In addition, the selected cutting speed can promote chatter or forced vibration that will chip the tool and/or reduce the quality of the produced surface.

Several investigators have simulated the physics of milling complex sculptured surfaces [92, 19, 5, 42, 96, 95, 84, 83, 79, 28, 29, 39, 103, 32, 9, 10, 1, 106, 54] . These simulations provide the following; the instantaneous chip geometry (chip load), the instantaneous cutting forces, and the static and dynamic tool deflections. The tool deflection and surface error are demonstrated in [32, 9, 10, 1, 106]. Optimizing the feed rate was addressed in [58, 21, 104]. Providing a representation of the tool was usually the first step presented in the above references. The second step was extracting the tool/workpiece interaction during cutting to determine chip loads, cutting forces, and tool deflections. The above authors have used two basic approaches for simulating the machining of complex sculptured surfaces. The first utilizes a solid modeling approach, the other takes an analytical approach.

The most common tool geometry used for machining complex sculptured surfaces is ball nose end mills [103, 32, 9, 10, 1, 106] . The geometry of other types of tool configurations such as flat, torus, and ball taper end mills were also simulated [108]. Both solid modeling and analytical modeling are used to simulate the complex tool geometry.

Solid modeling allows complex workpiece geometry and tool path locations to be simulated [79, 28, 29, 39] . A model of the material removed from the workpiece as a function of the tool geometry and tool path motion (usually called swept volume) can be developed much faster and easier using solid modeling than analytical modeling. Booleen operations are used to determine the intersection of the cutting edge representation in space and the swept volume at different time intervals. This intersection is called an in-cut segment and defines the instantaneous boundary of the chip geometry. The in-cut segment is multiplied by the feed per tooth to determine the chip load. Using solid modeling also allows a complex tool motion including tilt angles to be implemented. The efficient implemention of a solid modeling approach for sculptured surface machining requires a powerful spatial tool kit and/or a geometric engine. The commercially available spatial tool kits and geometric engines that can adequately simulate both complex tool and workpiece geometries are cost prohibitive to many industries. In addition, only global surface topology information is utilized to represent the workpiece using commercial systems. The global surface topology representations reduce the accuracy of the machining simulation. An enhancement for this type of modeling is to use local surface topology in the analysis. The ability to model the local surface topology of the workpiece is necessary to accurately determine the tool/workpiece interaction during machining. Modeling the local surface topology of the workpiece accounts for the material removed by previous tool paths in the vicinity of the current tool location.

Analytical modeling of the machining of complex sculptured surfaces offers a more robust methodology because it does not require a spatial tool kit or geometric engine to perform the simulation. Several analytical models for representing the tool geometry in space are available in the open literature [96, 79, 28, 29, 32, 9, 10, 1, 106, 54] . Analytical models for predicting the geometry of the instantaneous chip load are also available [39, 43]. Due to large computational times that are required to calculate the instantaneous chip load over the whole tool path, chip loads are only predicted for one or several revolutions of the tool. Cutting forces are then computed using mechanistic or semi empirical force models.

Most of these publications concentrate on one type of tool geometry, and a large percentage of them are developed for ball nose end mills. In the above research efforts the developed tool geometric model is only valid for the type of tool under investigation. In industry the machining of complex sculptured surfaces often necessitates the use of custom tool geometry. It is more practical in this case to develop a general analytical model that will be able to represent a large number of tools. Also, the predicted process variables (cutting force, temperature, deflection) are only simulated for a few revolutions of the tool in the above research. In practice, the process variables will change continuously during machining of a complex sculptured surface. It is, therefore, important to have an accurate prediction of the process variables at each point along the tool path so that optimization stategies can be reliably implemented (ie. feed rate scheduling).

2.4 Feed Scheduling

During machining of complex sculptured surface parts, the interaction between the cutting tool and workpiece varies continuously. This variation influences both the cutting force and tool deflection. In conventional machining, it is standard practice to set a constant feed rate based on the worst case cut geometry. While this practice provides guaranteed machining integrity (ie. moderate cutting forces, no tooth or cutter breakage, no chatter), the resulting machining time is not optimal. In order to obtain an optimal machining condition, both guaranteed machining integrity and minimum production time need to be considered. Two approaches have been used; in-process adaptive control, and pre-process planning. The following paragraphs identify the notable research work in these two areas.

Adaptive control strategies monitor the machining process using various sensors (force, vibration, etc.). Several types of adaptive control systems have been proposed; adaptive control optimization (ACO) [18, 97], adaptive control constraint (ACC) [31], and model reference adaptive control (MRAC) [87, 31, 73]. ACO designs attempt to optimize on-line a certain performance index, typically metal removal rate or machining costs, by manipulating the cutting speed and the feed rate [31]. ACC and MRAC systems are designed to regulate the cutting force, torque or power during machining. Regulation is achieved by adjusting the feed rate based on a transfer function relating the feed rate to the cutting force. Implementations of ACC and MRAC have been shown to offer better machine utilization and increased tool life.

During machining of complex sculptured surfaces, the cutting conditions are characterized by rapid changes in the cutting force (i.e depth of cut, feed rate) [99]. The performance of adaptive control systems is reduced when presented with severe transient phenomena (ie. sudden changes in axial depth of cut) [87, 24, 53, 83]. The response time of an adaptive control system is often large, resulting in poor control of the feed rate during transient phases [45].

If the interaction between the tool and workpiece can be accurately modeled prior to machining the part, a pre-process planning strategy can be utilized to achieve higher productivity and better product quality simultaneously. Force regulation is regarded as a practical way to satisfy safety, quality, and maximum metal removal rate criteria [6]. Force regulation can be achieved by adjusting the feed rate to maintain a constant maximum allowable force value. Feed rate scheduling is an example of an optimization strategy that can be implemented to ensure stable cutting during machining and/or to increase production rates. Several authors have considered this strategy [104, 21, 58, 34].

Yazar et al [104] offers a feed rate optimization strategy based on cutting force prediction in 3-axis milling of dies and molds with sculptured surfaces. The strategy first estimates the resultant cutting force at selected intervals along a specific cutter path. The maximum resultant cutting force along the path is used to schedule a feed rate that would not exceed a constraint force level. The cutting forces are then predicted again for that cutter path with the new feed rate. If the predicted forces are less than the constraint level force, the feed rate is increased to improve machining efficiency. The feed rate is adjusted based on a linear relationship between the resultant cutting force and the feed rate. The calculation of the cutting forces and the adjustment of the feed rate are carried out in an iterative way.

Chu et al [21] relates the feed rate selection to the local shape features during machining. These shape features includes plain, concave and convex corners, upward and downward ramps, and contour cutting. Machining tests investigating the geometric characteristics of the local shape features are carried out to establish a database for stable cutting conditions. A relationship between the local shape features and the optimal feed rate are developed based on this database. To maintain the cutting system integrity and reduce the cutting time, a post-processor is implemented to modify the feed rate specified by the G-Code file. The dynamics of the feed drive system are implicitly considered because the optimal feed rate is related to the local shape features obtained from machining experiments.

Lim et al [58] proposes a cutting path-adaptive feed rate strategy. A maximum feed rate map is used to select an optimum cutting direction and a feed rate to reduce machining time. A feed rate map for the desired part surface topology is defined by first dividing the part surface into a grid lattice of control points. At each control point, a maximum feed rate boundary is determined by applying either a force or dimensional constraint in different directions. This procedure is repeated for each control point. The distribution of the maximum feed rate boundaries is used to determine the cutting path direction. The intersection of the chosen cutting path direction and the control point feed rate boundary determines the scheduled feed rate.

Fussel et al [34] implements feed rate scheduling for multi-axis milling to maintain a constant force level. The proposed feed rate process planner is based on an integrated mechanistic and geometric end milling model. An iterative solution is used to solve for the optimal feed rate at each step along a tool path. The mechanistic model first estimates the force produced by a "first guess" feed rate value. Using a bisection method, the feed rate is adjusted and the mechanistic model is invoked again. The iterations continue until the predicted force is within a given tolerance range of the target force value. This procedure is repeated for each tool move in the G-Code file, resulting in a set of optimal feed rate.

2.5 Internet Based Machining Systems

The Internet is a global information system that provides a communications medium for connecting computers, and computing resources around the world. Since its early inception in the 1970's, the internet has grown in size and popularity to provide many services and applications. The major internet services and applications include: electronic mail, file transfer protocols, electronic discussion groups, search tools, and the world wide web, a hypertext based information medium. Commercial entities have also begun to leverage the internet for various forms of e-commerce and/or e-business. In the area of engineering, the internet offers a medium for technology integration. This involves both information exchange and providing access to engineering resource tools. Notable research and development efforts in this area have included those of the Machine Tool Agile Manufacturing Research Institute (MTAMRI) and University of California- Berkeley.

The MTAMRI is an inter-university collaborative research group that has developed a series of software testbeds for modeling machining systems. A software testbed is simply defined as a web-based interface to an application program. The software testbeds that have been developed by MTAMRI include: EMSIM, TAPSIM, DRSIM, FIXMA, and CFEST. EMSIM is a software application that simulates an end milling machining process [26, 25]. The user inputs for the EMSIM simulation include: cut geometry, workpiece material, end mill geometry, and machining conditions (feeds and speeds). The outputs of the simulation include cutting forces, cutting power, and part/cutter deflections. TAPSIM simulates the cutting forces (torque and thrust) in the tapping process [65]. The simulation software requires input information on the tap geometry, workpiece type, and process parameters. The simulation output includes thrust forces and torque. DRSIM simulates the cutting forces (torque and thrust) in the drilling process [65]. The simulation input includes the drill geometry, workpiece type and process parameters. The simulation output includes thrust and radial forces and torque. FIXMA is a fixture design and analysis simulation package [25]. It integrates process modeling, fixture analysis, and fixture optimization. CFEST is a cutting fluid evaluation toolkit [82]. The user inputs information related to cutting fluid type, application method, machining parameters and site specific factors. The output gives quantitative information related to the cutting fluid's process performance, environmental impact, health and safety hazard scoring, and the costs for procurement as well as for waste disposal treatment.

The University of California-Berkeley has developed an internet-accessible, integrated CAD/CAPP/CAM development environment termed CyberCut [75, 101]. A feature-based, constrained destructive solid geometry CAD environment is utilized by CyberCut. The concept here is to create a part from a prismatic stock by removing feature primitives. Feature primitives are defined by conventional machining operations and include pockets, holes, arcs and face-off operations. Although the part design may be limited by the available feature primitives, the part design is deterministic with respect to manufacturability. The Cybercut CAPP system can be utilized in either an incremental or batch mode. When used in incremental mode, the manufacturing plan is updated after each feature is added to the part design. A computationally intensive knowledge-based CAPP system is used here. The knowledge-based CAPP system is accessed remotely and does not reside on the client's computer. The batch mode process planner uses client-side processing. A set of design rules specific to the fabricating machine is uploaded from the fabrication site. The designer uses this set of rules to create the part. Once the part has been designed and a process plan completed, the information is passed to a fabrication site where a prototype part is produced.

2.6 Summary

The scope of this thesis includes research areas related to process modeling, process optimization and Internet technologies for machining systems. This chapter provided a detailed overview of these research areas.

The review of process modeling methods reveals several key elements necessary for developing a generic approach for simulating multi-axis machining. The modeling methodology should offer a generalized approach for representing the cutting edge, the local surface topology of the part, and the tool/part interaction during machining. The force modeling methodology should be developed for multi-axis machining. The force model calibration technique used should incorporate the effects of varying cutting edge geometry, cutting speeds, and feed rates. The modeling approach should also include process dynamics.

The review of feed scheduling revealed two basic approaches; in-process adaptive control, and pre-process planning. The performance of adaptive control systems is reduced for complex sculptured surface machining where the cutting conditions are characterized by rapid changes in the cutting conditions. The pre-process planning approach for feed scheduling can be applied for complex sculptured surface machining provided the tool/workpiece interaction can be accurately modeled. The machine tool feed drive performance capabilities were not integrated with feed scheduling in any of the reviewed literature.

The review of Internet based machining system reveals the potential for leveraging existing internet technologies to create a virtual process simulation and optimization environment.

Chapter 3

Process Modeling

3.1 Introduction

This chapter presents the modeling methodologies for a generic simulation approach for multi-axis machining. To accurately simulate the machining process for complex sculptured surface parts, a comprehensive geometric modeling scheme and cutting force model are required. The geometric modeling scheme must offer a generalized approach for representing the tool cutting edge, the local surface topology of the part, and the tool/part interaction during machining. The geometric modeling scheme is presented in section 3.2. The cutting force modeling methodology for multi-axis machining is given in section 3.3. The cutting force model requires a set of calibration coefficients. A systematic model calibration procedure that incorporates the effects of varying cutting edge geometry, cutting speeds, and feed rates is developed in section 3.4. A methodology for incorporating tool runout is described in section 3.5. A description of the implementation procedure for the modeling methodologies presented in sections 3.2 and 3.3 for simulating multi axis machining of complex sculptured surfaces is presented in section 3.6. Two methodologies are presented for simulating process dynamics in multi-axis machining. A piece-wise continous solution based on tool path control points is given in section 3.7. A continous solution for multi-axis machining is presented in section 3.8. The concept and implementation of an Internet based facility for multi-axis milling process simulation and optimization is described in section 3.9. A summary is given in section 3.10.

3.2 Geometric Modeler

3.2.1 Introduction

The key elements of the machining process that must be considered when formulating a geometric modeling scheme for multi-axis machining include the representations for the tool cutting edge, the local surface topology of the part, and the tool/part interaction during machining. The following sections provide an overview of these key elements including details related to their implementation.

3.2.2 Cutting Edge Representation

The primary type of tooling used today for multi-axis machining of complex sculptured surfaces include ball and flat end mills. However, the specific cutting edge design varies significantly within this class of tooling. In addition, custom tooling is often required. Therefore, it is important to have a generalized modeling procedure to represent this varied cross-section of tooling. The majority of the literature has dealt with simple cutting edge designs (ie. zero helix, constant helix, constant pitch)[77].

Only recently has the literature presented work that attempts to generalize the representation of the cutting edge. Imani [39] and Abrari [3] use a cubic piecewise NURBS curve to represent the tool cutting edge profile for a constant lead /pitch ball end mill. A pth degree NURBS curve is defined by [69]

$$C(u) = \frac{\sum_i^n N_{i,p}(u)\omega_i P_i}{\sum_i^n N_{i,p}(u)\omega_i} \tag{3.1}$$

where the P_i are the control points, the ω_i are the weights and the $N_{i,p}(u)$ are the pth B-spline basis functions defined on the non-periodic knot vector

$$U = \{\underbrace{a,\ldots,a}_{p+1}, u_{p+1},\ldots,u_{m-p-1},\underbrace{b,\ldots,b}_{p+1}\} \tag{3.2}$$

The ith B-spline basis functions of degree p, denoted by $N_{i,p}(u)$, is defined as

$$N_{i,0}(u) = \begin{cases} 1 & if\ u_i \le u \le u_{i+1} \\ 0 & otherwise \end{cases} \tag{3.3}$$

$$N_{i,p}(u) = \frac{u-u_i}{u_{i+p}-u_i}\cdot N_{i,p-1}(u) + \frac{u_{i+p+1}-u}{u_{i+p+1}-u_{i+1}}\cdot N_{i+1,p-1}(u) \tag{3.4}$$

The goal here is to extend this work to include NURBS curve representations of an arbitrary cutting edge design. To accomplish this goal, a generalized modeling procedure is necessary. The following steps outline the proposed procedure:

1. For a given cutting edge design, a set of 3D points are generated along the cutting edge profile. There are several plausible methods for generating these 3D points; an analytical expression for the edge profile, a tooling drawing, or

discrete 3D points measured by a Coordinate Measuring Machine (CMM).

2. Interpolate the 3D points with a NURBS curve. The curve fitting algorithm used here interpolates the 3D points precisely.
3. If the 3D points are generated by a CMM, the data usually contain measurement and computational noise. To avoid this noise, the user may want to approximate the 3D points with a NURBS curve. Facilities are provided to approximate a NURBS curve to 3D points.

A generalized NURBS curve form can now be used to represent an arbitrary cutting edge profile. The primary advantage of representing the cutting edge profile as a NURBS curve is that there are standard techniques available for manipulating the location and orientation of a NURBS curve. This manipulation is necessary for multi-axis machining, where the orientation of the cutting edge changes continuously along a specified tool path.

Implementation

A set of software modules (written using the $C++$ language) were used to implement the facilities for constructing a generalized representation of the cutting edge. A cubic piecewise NURBS curve is used to represent an arbitrary cutting edge design. Figure 3.1 depicts the major software routines used. The 3D points are generated from either a CMM, analytic expression of the cutting edge, or tool drawing. The type of cutting edge designs that are currently supported include:

1. straight inclined edge (ball/flat end mill)
2. constant helix flat end mill

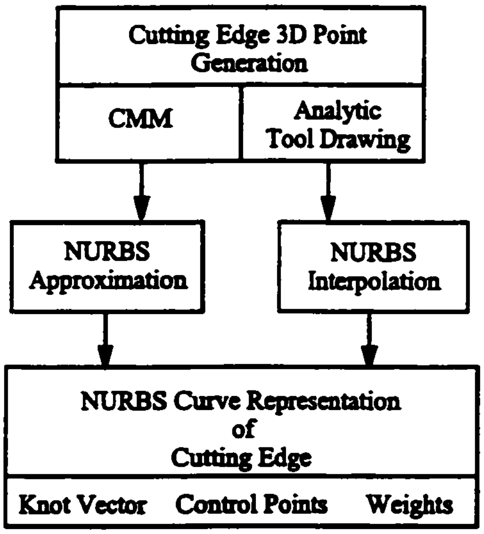


Figure 3.1: Software Routines for NURBS Representation of Cutting Edge

3. constant lead helical ball end mill
4. constant helix taper ball end mill
5. custom edge design (points provided by CMM)

The user is then presented with the option of using either a global interpolation routine or global approximation routine to fit a NURBS curve to the set of 3D points. The global interpolation routine is used to precisely fit the data points. The approximation routine is used to fit a smooth NURBS curve to data that has either measurement or computational noise. The approximation routine is typically used to interpolate points provided by a CMM.

3.2.3 Local Surface Topology

To accurately model the tool/part interaction during machining of a part composed of many complex sculptured surfaces, the local surface topology of the part must be represented. The local surface topology of the part is defined here as those surfaces generated by previous tool paths in the vicinity of the current tool position. Commercially available CAD systems are effective at representing the global surface topology of the part. However, their representation of local surface topology (as defined here) is limited and at best computationally intensive. The concept here is to offer a generalized approach to representing the local surface topology of the part without using a computationally expensive CAD system. The proposed methodology is as follows. The local surface topology will be represented by only those surfaces generated by previous tool paths in the vicinity of the current tool position. The surface generated by each tool path will be represented as a set of geometric primitives (ie. planes, cylinders, spheres, torus). The procedure for generating the local surface topology of the part will be:

1. Extract the tool path information from the given CLData files (roughing, semi-finishing, finishing).
2. Generate a data structure for each tool path segment. This data structure contains the primitive surface representation and a bounding box for the given tool path segment.
3. During simulation, a control space is defined at the current tool position. A control space is defined here as a bounding box surrounding the current tool position. The size of the control space is defined by the user and is related to

the geometry of the tool.

4. A simple bounding box intersection algorithm is used to determine which tool path segments contribute to the local surface topology of the part in the vicinity of the current tool position.

The advantage of the above methodology is that all the information required to create an accurate representation of the local surface topology of the part is contained within simple data structures for each tool path segment. For a complex sculptured surface, the number of tool path segments may be large as well as the required memory storage for each tool path data structure. However, this memory storage is held static during a simulation and each tool path data structure is pre-computed prior to running the simulation.

Implementation

The software module that generates the local surface topology performs the following tasks:

1. Parses the given CLData file to extract individual tool path segments

2. Creates a data structure for each tool path segment. This data structure contains the primitive surface representation (cylinder, plane, sphere, torus) and a bounding box for the given tool path segment (see figure 3.2)

3. Defines a control space for the current tool position and determines which tool path segments contribute to the surface topology of the control space. Only those tool path segments which intersect the control space are used to create the local surface topology (see figure 3.3). A simple bounding box intersection

algorithm is used here to efficiently extract the set of tool path segments that contribute to the local surface topology.

3.2.4 Tool/Part Interaction

Determining the tool/part interaction is necessary to define the cutting edge incut segments. Figure 3.4 shows how the incut segment varies for semi-finishing and finishing machining operations. These incut segments define the integration limits for a multi-axis milling force model. The proposed methodology for determining the cutting edge incut segments is:

For a defined local surface topology and angular position of the cutting edge

1. For each point along the cutting edge, classify as either in/out of each tool path bounding box which defines the local surface topology.
2. If the point on the cutting edge is contained within a tool path bounding box, determine the corresponding Z value from the primitives representing that surface ($Z_{surf,i}$).
3. Find the minimum of all $Z_{surf,i}$ (Z_{min}).
4. If the cutting edge point Z value is less than Z_{min} then the cutting edge is incut.

The above methodology is used to extract both the start and end of incut segment.

Implementation

The software module that determines the cutting edge incut segments classifies each point along the cutting edge against the local surface topology of the part. If the

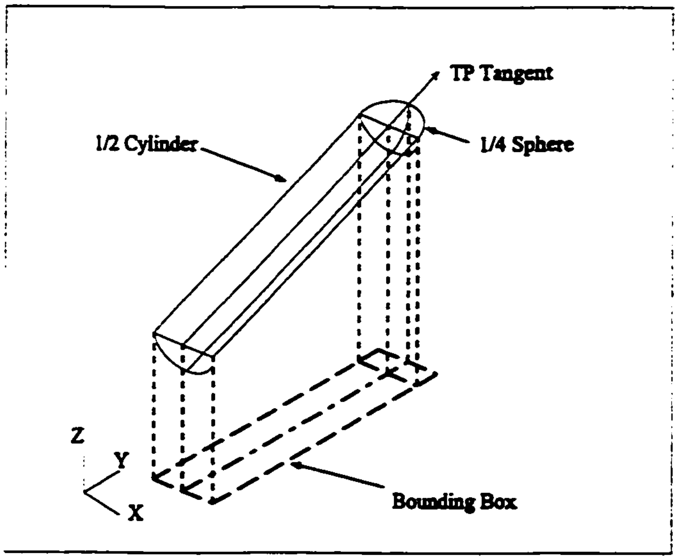


Figure 3.2: Tool Path Representation

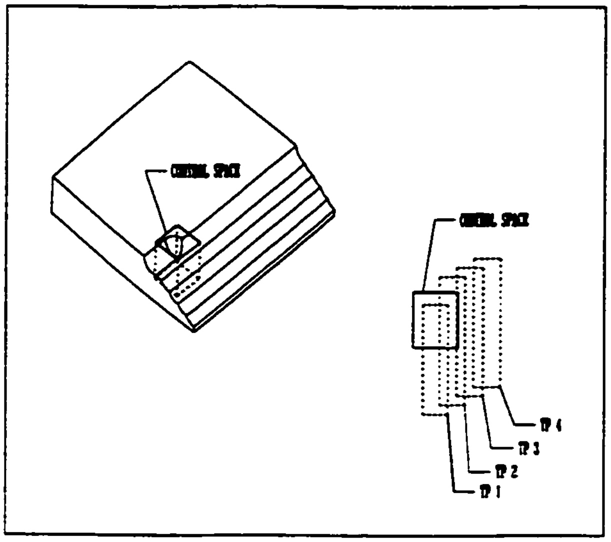


Figure 3.3: Control Space Definition

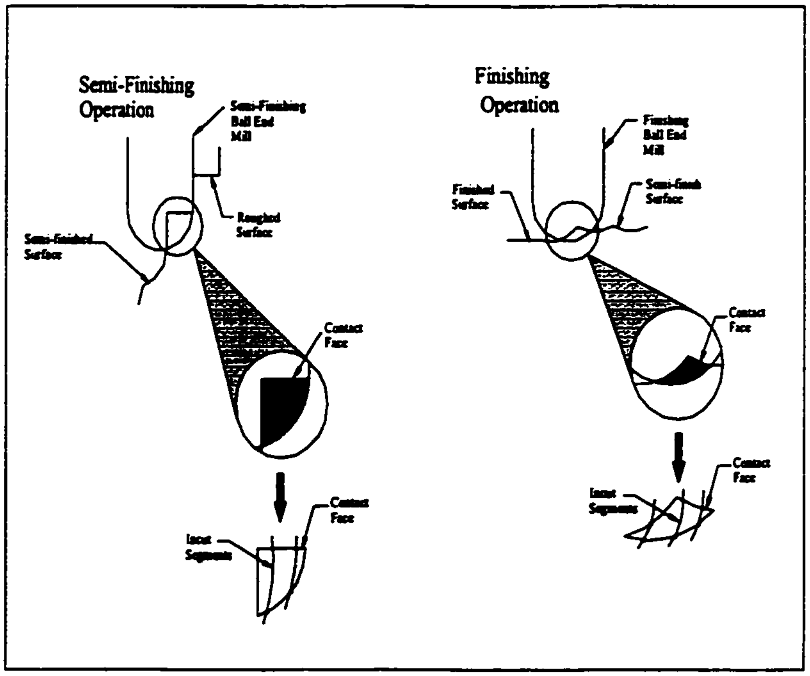


Figure 3.4: Variation of InCut Segments (semi-finishing, finishing)

current point on the cutting edge is below the part surface, it is incut. If the current point on the cutting edge is above the part surface, it is out of cut. This classification scheme defines the incut segments for the cutting edge at each incremental angular position of the tool. The incut segments are used as the integration limits for the multi-axis milling force model. Figure 3.5 and 3.6 show incut segment limits for 1/2 and 1/4 immersion ball end milling.

3.3 Force Modeler

A mechanistic force model for multi-axis machining is presented in this section. This model has been formulated similar to the models presented by Imani [39] and Yucesan

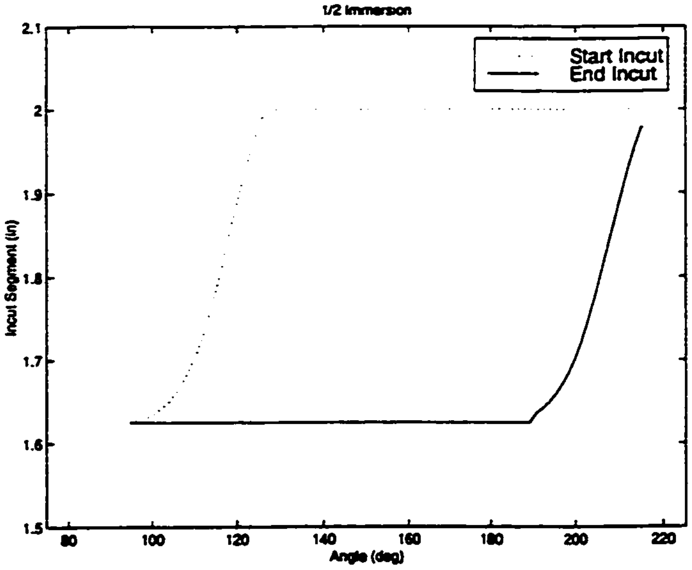


Figure 3.5: 1/2 Immersion InCut Segment Limits

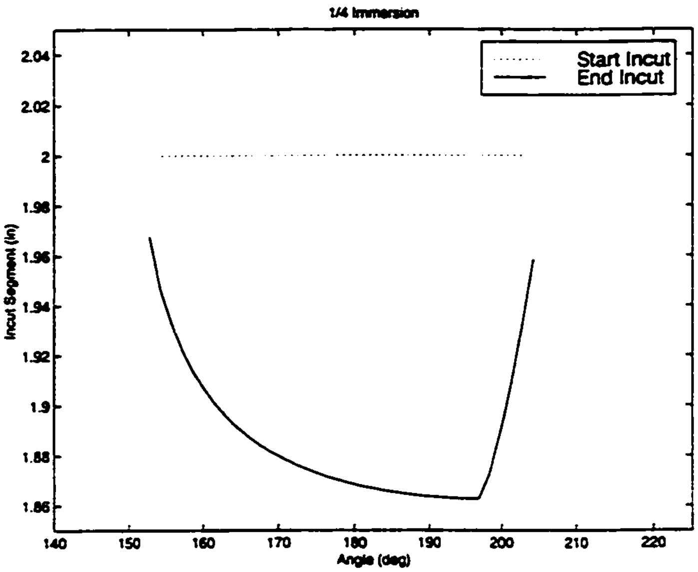


Figure 3.6: 1/4 Immersion InCut Segment Limits

[105]. The modeling approach taken was to first develop the cutting edge geometry, then to develop the cutting force expressions.

The cutting edge geometry for a ball end mill is shown in figure 3.7. The orientation of the differential cutting forces on the rake and clearance face are determined by the unit vectors shown in the figure. The derivation of these unit vectors follows from the defined NURBS curve profile of the cutting edge, and the nominal rake, clearance, and helix angles for the cutting edge. The nomenclature used will be defined throughout the derivation.

At any point, P_i, along the cutting edge, the coordinate values can be determined from the NURBS curve representation, given the corresponding parameter u_i. The parameter, u_i, is extracted from a parameter set, $\{U\}$. This parameter set gives u_i having equal z-axis spacing. The unit vector tangent to the cutting edge flute, $\vec{\tau}$, is defined as

$$\vec{\tau} = \frac{\vec{P_i} - \vec{P}_{i-1}}{\left\|\vec{P_i} - \vec{P}_{i-1}\right\|} \tag{3.5}$$

where $\vec{P_i}$ and $\vec{P}_{i-1}$ are vectors defined by consecutive points on the cutting edge (P_i, P_{i-1}) and the origin of the tool. The instantaneous cutter helix angle, β_z, is determined from the angle between $\vec{\tau}$ and a unit vector along the direction of the tool z-axis ($\vec{Z} = \langle 0, 0, 1\rangle$),

$$\beta_z = \cos^{-1}(\vec{\tau} \cdot \vec{Z}) \tag{3.6}$$

The lag angle, φ, is measured from the cutting edge tip (at parameter value, u_i) in a

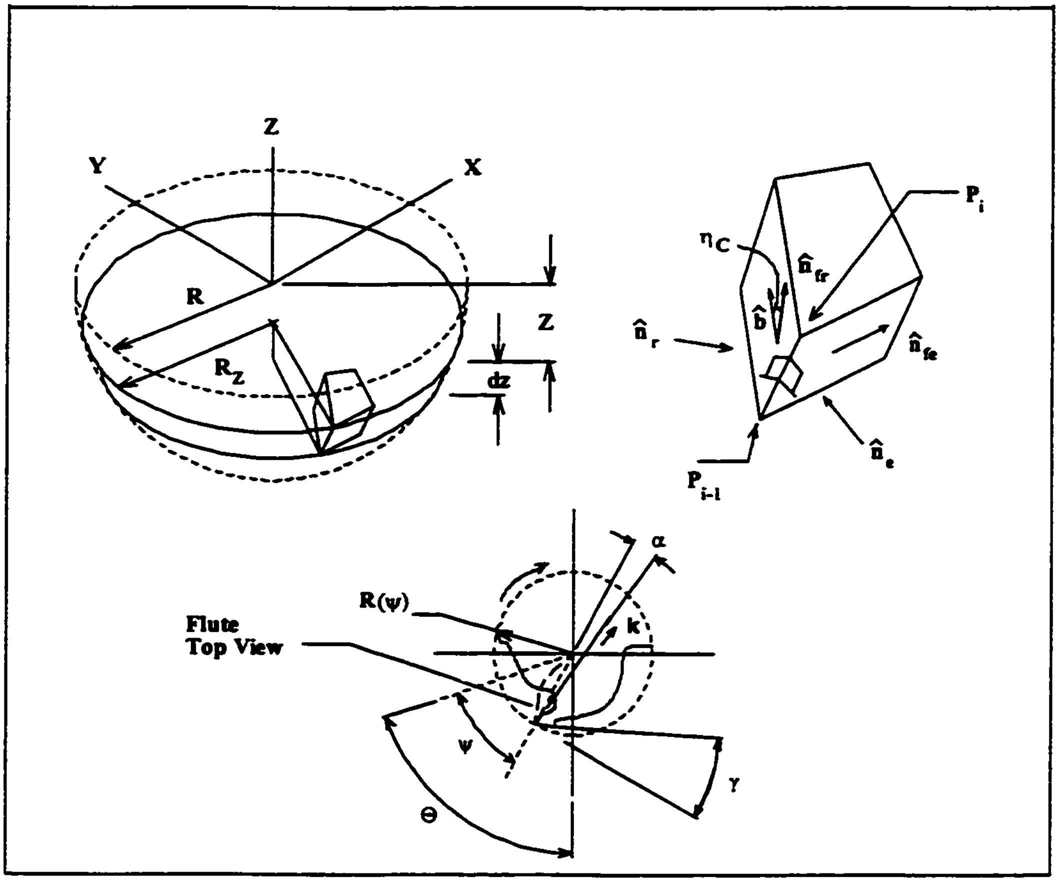


Figure 3.7: Ball End Mill Geometry

CCW direction. This angle lies in the XY plane (see figure 3.7).

$$\varphi = \cos^{-1}(\vec{P}_i \cdot \vec{P}_0)_{z=0} \tag{3.7}$$

The instantaneous angle of rotation, at P_i, is θ_i. It is defined from the angle of rotation at the cutting edge tip, θ_0, and the lag angle, φ.

$$\theta_i = \theta_0 - \varphi \tag{3.8}$$

The lead angle, ρ_i, is defined by the nominal radius of the tool, R_0, and the current z position, z_i,

$$\rho_i = \sin^{-1}\left(\frac{z_i}{R_0}\right) \tag{3.9}$$

The instantaneous normal rake angle, α_i, is determined from the nominal rake angle, α_0, the lead angle, ρ_i, and the instantaneous cutter helix angle, β_z

$$\alpha_i = \tan^{-1}([\tan\beta_z \cdot] \sin\rho_i + \tan\alpha_0 \cdot \cos\rho_i] \cdot \cos\beta_z) \tag{3.10}$$

Determine κ. This unit vector lies on the rake face and along the intersection of a parallel xy-plane at the cutting edge point that meets the rake surface (see figure 3.7). A unit vector,$\vec{V}_u$, is defined as,

$$\vec{V}_u = \frac{\vec{P}_{i,z=0}}{\|\vec{P}_{i,z=0}\|} \tag{3.11}$$

$\vec{V}_u$ is rotated by the instantaneous rake angle, α_i, to determine κ,

$$\begin{aligned} \kappa_x &= V_{u,x} \cdot \cos(-\alpha_i) - V_{u,y} \cdot \sin(-\alpha_i) \\ \kappa_y &= V_{u,x} \cdot \sin(-\alpha_i) - V_{u,y} \cdot \cos(-\alpha_i) \end{aligned} \tag{3.12}$$

The unit vector that is normal to the rake face, $\vec{n}_r$, is computed as the cross product of the vectors $\vec{\tau}$ and κ.

$$\vec{n}_r = \vec{\tau} \times \kappa \tag{3.13}$$

The unit vector, $\vec{b}$, that lies on the plane created by the vectors $\vec{\tau}$ and κ, and is perpendicular to $\vec{\tau}$, is determined from the cross product of $\vec{\tau}$ and $\vec{n}_r$.

$$\vec{b} = \vec{n}_r \times \vec{\tau} \tag{3.14}$$

The unit vector on the rake face that defines the direction of chip flow is $\vec{n}_{fr}$. $\vec{n}_{fr}$ is computed from $\vec{b}$, $\vec{\tau}$, and the chip flow angle, η_c,

$$\vec{n}_{fr} = \vec{b} \cdot \cos\eta_c + \vec{\tau} \cdot \sin\eta_c \tag{3.15}$$

Determine ξ. This unit vector lies on the clearance face and along the intersection of a parallel xy-plane at the cutting edge point that meets the clearance surface (see figure 3.7). A unit vector, $\vec{T}_u$, is defined as,

$$\vec{T}_u = \perp\vec{V}_u \tag{3.16}$$

$\vec{T}_u$ is rotated by the instantaneous clearance angle, γ_i, to determine ξ,

$$\begin{aligned} \xi_x &= T_{u,x} \cdot \cos(\gamma_i) - T_{u,y} \cdot \sin(\gamma_i) \\ \xi_y &= T_{u,x} \cdot \sin(\gamma_i) - T_{u,y} \cdot \cos(\gamma_i) \end{aligned} \tag{3.17}$$

The unit vector that is normal to the clearance face, $\vec{n}_e$, is computed as the cross product of the vectors $\vec{\tau}$ and ξ.

$$\vec{n}_e = \vec{\tau} \times \xi \tag{3.18}$$

The unit vector, $\vec{n}_{fe}$, that lies on the plane created by the vectors $\vec{\tau}$ and ξ, and is perpendicular to $\vec{\tau}$, is determined from the cross product of $\vec{\tau}$ and $\vec{n}_e$.

$$\vec{n}_{fe} = \vec{n}_e \times \vec{\tau} \tag{3.19}$$

The length of an infinitesimal cutting edge segment, δS, is given as

$$\delta S = \|\vec{P}_i - \vec{P}_{i-1}\| \tag{3.20}$$

The instantaneous chip thickness, t_c, at any angular position, θ_i is given by

$$t_c(\theta) = s_t \sin(\theta) \cos(\lambda) \cdot (\kappa \cdot \vec{b}) \tag{3.21}$$

where s_t is the nominal feed per tooth and λ is the current tool path inclination angle. The differential contact area is given as,

$$\delta A_r = t_c(\theta) \cdot \delta S \tag{3.22}$$

The force model expressions are considered as the summation of the normal and friction forces acting on the rake face and the cutting edge. The differential normal and frictional forces on the rake face are calculated as

$$|\delta \Im_{nr}| = K_{nr} \cdot \delta A_r \tag{3.23}$$

$$|\delta \Im_{fr}| = K_{fr} \cdot \delta A_r \tag{3.24}$$

where K_{nr} and K_{fr} are the pressure and friction cutting coefficients on the rake face.

These coefficients are functions of cutting edge geometry, cutting speed, and feed per tooth.

The differential normal and frictional edge forces are calculated as

$$
\begin{aligned}
|\delta\Im_{ne}| &= K_{ne}\cdot\delta S &\quad (3.25)\\
|\delta\Im_{fe}| &= K_{fe}\cdot\delta S &\quad (3.26)
\end{aligned}
$$

where K_{ne} and K_{fe} are the pressure and friction cutting coefficients acting on the cutting edge. These coefficients are functions of cutting edge geometry, cutting speed, and feed per tooth.

The instantaneous force acting on an incut, cutting edge segment is given as the summation of the differential normal and friction forces acting on the rake face and cutting edge.

$$
\begin{aligned}
\vec{\Im}_p(\theta) &= \vec{\Im}_{nr}(\theta)+\vec{\Im}_{fr}(\theta)+\vec{\Im}_{ne}(\theta)+\vec{\Im}_{fe}(\theta)\\
&= \int_{z1}^{z2}\{K_{nr}\cdot[-\vec{n}_r(\theta,z)+K_{fr}\cdot\vec{n}_{fr}(\eta_c,\theta,z)]\cdot\delta A_r \qquad (3.27)\\
&\quad +K_{ne}\cdot[-\vec{n}_e(\theta,z)+K_{fe}\cdot\vec{n}_{fe}(\theta,z)]\cdot\delta S\}
\end{aligned}
$$

3.4 Calibration Procedure

Calibration tests are used to determine the friction and pressure cutting force coefficients. It is proposed that these coefficients are functions of cutting edge geometry, cutting speed, and feed per tooth. The functional form of the equations representing the variation of pressure and friction coefficients will take the form

$$K_i = e^{a_1 + a_2 V_j + a_3 \beta_j} \tag{3.28}$$

where K_i is the instantaneous cutting coefficient, V_j and β_j are the instantaneous cutting speed and geometric constant respectively. The coefficients a_1, a_2, a_3 are functions of the instantaneous feed per tooth. The functional form of equation 3.28 was formulated similar to Kapoor *et al*[43]. Kapoor expressed the cutting force coefficients as an exponential function of chip thickness and cutting speed. Kapoor did not incorporate the effect of cutting edge geometry.

Following the theoretical derivation of the multi-axis force model in section 3.3, a calibration procedure was developed. The calibration procedure is as follows:

1. The first step in the calibration procedure is to reformulate the multi-axis force model presented in section 3.3 into a form that separates the cutting force coefficients and geometric factors.

$$\begin{aligned} \vec{\Im}_p(\theta) &= \vec{\Im}_{nr}(\theta) + \vec{\Im}_{fr}(\theta) + \vec{\Im}_{ne}(\theta) + \vec{\Im}_{fe}(\theta) \\ &= \int_{z1}^{z2} \{K_{nr} \cdot [-\vec{n}_r(\theta, z) + K_{fr} \cdot \vec{n}_{fr}(\eta_c, \theta, z)] \cdot \delta A_r \\ &\quad + K_{ne} \cdot [-\vec{n}_e(\theta, z) + K_{fe} \cdot \vec{n}_{fe}(\theta, z)] \cdot \delta S\} \end{aligned} \tag{3.29}$$

if $\vec{n}_{fr}(\eta_c, \theta, z) = \vec{b}(\theta, z) \cdot \cos\eta_c + \vec{r}(\theta, z) \cdot \sin\eta_c$

$$
\begin{aligned}
\vec{\Im}_p(\theta) \;=\; & \int_{z1}^{z2} \{K_{nr} \cdot [-\vec{n}_r(\theta, z) + K_{fr} \cdot (\vec{b}(\theta, z) \cdot \cos \eta_c \\
& + \vec{\tau}(\theta, z) \cdot \sin \eta_c)] \cdot \delta A_r \\
& + K_{ne} \cdot [-\vec{n}_e(\theta, z) + K_{fe} \cdot \vec{n}_{fe}(\theta, z)] \cdot \delta S\}
\end{aligned}
\tag{3.30}
$$

Rearranging using:

$$
K_r = \begin{bmatrix} K_{nr} \\ K_{nr} \cdot K_{fr} \cdot \cos \eta_c \\ K_{nr} \cdot K_{fr} \cdot \sin \eta_c \end{bmatrix} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix} \tag{3.31}
$$

$$
K_f = \begin{bmatrix} K_{ne} \\ K_{ne} \cdot K_{fe} \end{bmatrix} = \begin{bmatrix} K_4 \\ K_5 \end{bmatrix} \tag{3.32}
$$

$$
\hat{R}(\theta, z) = \begin{bmatrix} -n_{rx} & b_x & \tau_x \\ -n_{ry} & b_y & \tau_y \\ -n_{rz} & b_z & \tau_z \end{bmatrix} \tag{3.33}
$$

$$
\hat{E}(\theta, z) = \begin{bmatrix} -n_{ex} & n_{fex} \\ -n_{ey} & n_{fey} \\ -n_{ez} & n_{fez} \end{bmatrix} \tag{3.34}
$$

$$
\vec{\Im}_p(\theta) \;=\; \int_{z1}^{z2} \{\hat{R}(\theta, z) \cdot K_r \cdot \delta A_r(\theta, z) + \hat{E}(\theta, z) \cdot K_f \cdot \delta S\} \tag{3.35}
$$

It can be shown that equation 3.35 can also be represented in the form (see Appendix A)

$$\begin{aligned} \Im_{px} &= K_1 \cdot A_1 + K_2 \cdot A_2 + K_3 \cdot A_3 + K_4 \cdot A_4 + K_5 \cdot A_5 \\ \Im_{py} &= K_1 \cdot B_1 + K_2 \cdot B_2 + K_3 \cdot B_3 + K_4 \cdot B_4 + K_5 \cdot B_5 \\ \Im_{pz} &= K_1 \cdot C_1 + K_2 \cdot C_2 + K_3 \cdot C_3 + K_4 \cdot C_4 + K_5 \cdot C_5 \end{aligned} \qquad (3.36)$$

Equation 3.36 is in the form of a multiple parameter, linear regression model.

2. A series of cutting tests are performed using the chosen tool/workpiece combination. These tests are performed over a range of cutting speeds, feed rates, axial depth of cut, and radial width of cut. For each cutting test, cutting force data is obtained ($\Im_{mx}$, $\Im_{my}$, $\Im_{mz}$).

3. For each cutting test, average cutting force coefficients [K_1, K_2, K_3, K_4, K_5] are determined using a multiple parameter linear regression technique (Least-Square Estimation) (see Appendix B).

4. It is assumed that for constant feed rate, each K_i can be represented in the form given by equation 3.28. Taking the natural logarithm of equation 3.28 gives:

$$\ln K_i = a_1 + a_2 \cdot V_j + a_3 \cdot \beta_j \qquad (3.37)$$

Grouping those tests that have constant feed rate, the parameters a_i are determined for each K_i using multiple parameter log-linear regression (see Appendix C).

5. For each K_i, the parameters a_1, a_2, and a_3 are expressed as a function of the feed rate as:

$$a_i = b_i + c_i f_i \tag{3.38}$$

where a_i is the chosen parameter, b_i is the y-intercept of a best fit line, c_i is the slope of a best fit line, and f_i is the instantaneous feed per tooth. Equation 3.38 expresses each parameter, a_i, as a linear function of feed per tooth.

3.5 Tool Runout

Tool runout affects the cutting force profile for each cutting edge during machining. The tool runout acts to increase or decrease the effective feed per tooth for each cutting edge. Several investigators have modeled tool runout [94, 109, 108]. Tool runout has been identified as either radial, axial, or combinational. Tool runout can also be non-linear due to secondary disengagement resulting from insufficient depths of cut. Radial effects have been shown to contribute the most to tool runout. The tool runout also affects the incut segments for each cutting edge.

Radial tool runout has been included in the modeling methodology presented here. The tool runout is taken as a user supplied input parameter for each cutting edge. Tool runout is quantified as a radial displacement relative to the nominal radius of the tool. The tool runout can be either positive or negative.

The effective feed per tooth for each cutting edge, $s_{t,i}$, is determined by summing the nominal feed per tooth, s_t, and cutting edge runout, r_i.

$$s_{t,i} = s_t + r_i \tag{3.39}$$

To include the effects of tool runout on the determination of each cutting edge incut segment, the NURBS curve representation of each tool cutting edge is translated a distance equal to the cutting edge runout. The translation is in the tool radial direction.

A set of simulations were performed to demonstrate the effects of different tool runout on the cutting force. All simulations are performed with a four flute 330M Garr 3/4" diameter, solid carbide ball end mill. The workpiece material was Titanium (Ti6Al4V). The feed per tooth is 0.004" and the spindle speed is 1176 rpm. The immersion ratio is 1/8. The force calibration constants are given in table 3.1.

Rake face			Clearance Face	
K_1 ($lb/inch^2$)	K_2 ($lb/inch^2$)	K_3 ($lb/inch^2$)	K_4 ($lb/inch$)	K_5 ($lb/inch$)
152176	199736	3610	39.06	26.18

Table 3.1: Force Calibration coefficients

Figures 3.8 and 3.9 demonstrate the effect of different tool runout on the resultant cutting force. The resultant force profile is shown in each figure. The solid line in each figure is the predicted resultant force with runout. The dotted line in each figure is the predicted resultant force without runout. Table 3.2 shows the measured tool runout for each figure.

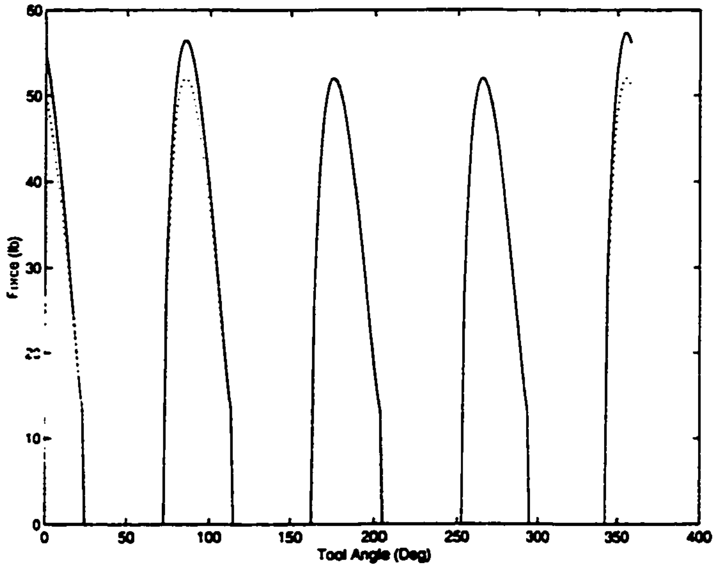


Figure 3.8: Runout 1 (resultant force)

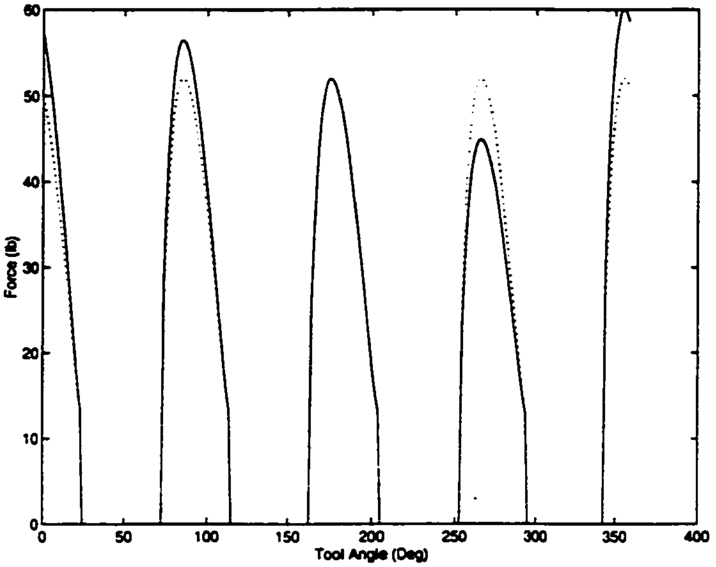


Figure 3.9: Runout 2 (resultant force)

Figure	Edge 1	Edge 2	Edge 3	Edge 4
3.8	0.0	+0.0005	+0.0006	0.0
3.9	0.0	+0.0005	+0.0009	-0.0008

Table 3.2: Tool Runout

3.6 Static Simulation Methodology

The implementation for the methodology for analytically simulating multi axis machining of complex sculptured surfaces is shown in figure 3.10.

The simulation begins by reading the tool path information from a CLData file. For each tool path segment a data structure is created. This data structure contains the primitive surface representation (cylinder, plane, sphere, torus) and a bounding box for the given tool path segment (figure 3.2). A set of control points are defined at a user specified interval along the tool path. At each control point the following steps are performed:

1. A control space is defined for the current tool position. A control space is defined as a bounding box surrounding the current tool position (figure 3.3). The size of the control space is defined by the user and related to the geometry of the tool.

2. The local surface topology is constructed for the current tool position. The local surface topology is defined by previous tool path segments that intersect the control space.

3. An instance of the tool cutting edge is created and translated to the current tool position.

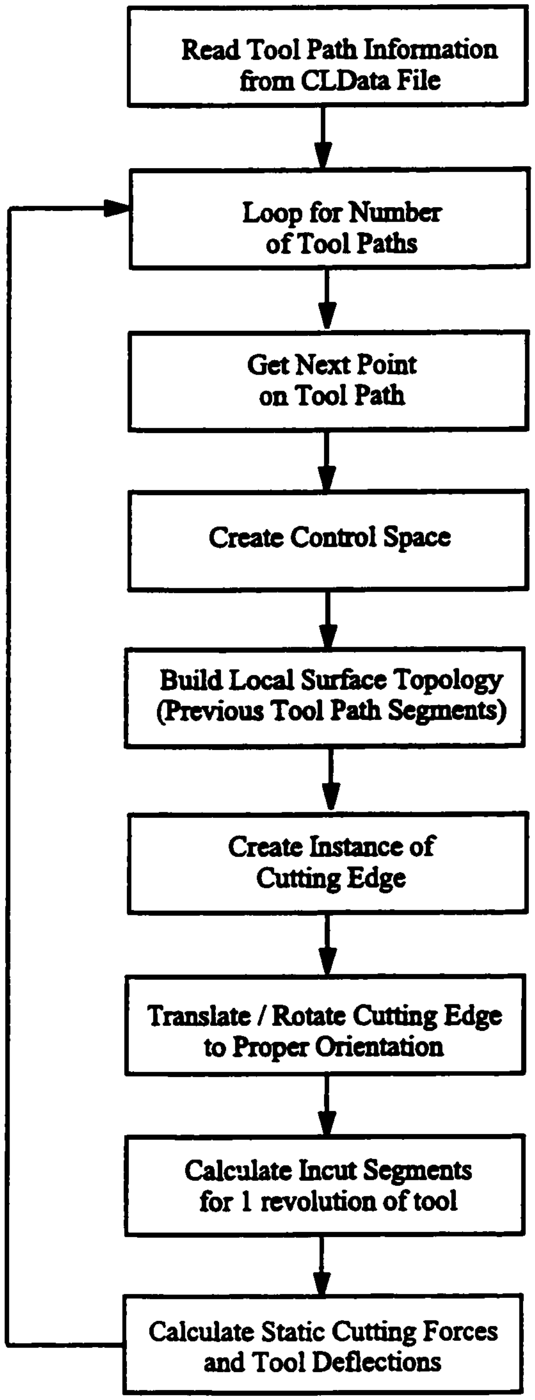


Figure 3.10: Static Modeling Flow Chart

4. The tool immersion is determined. The tool immersion is defined by the cutting edge incut segments ($ICSeg$) at each angular increment of the tool. The instance of the tool cutting edge is incrementally rotated through one full revolution of the tool. At each angular increment the cutting edge incut segment is determined by classifying points on the cutting edge against the local surface topology.

5. The cutting force is calculated at each angular increment of the tool. Figure 3.11 shows a flow chart for the static force prediction. For each angular increment, i, and each cutting edge, j, the incut segment, $ICSeg[i][j]$, is used as the integration limits for the force prediciton. For each incut segment the cutting edge is discretized into a number of axial elements, Ax_{inc}

   $$Ax_{inc} = \frac{ICSeg[i][j]}{dz} \tag{3.40}$$

   where dz is the minimum axial distance. For each axial element the following are determined:

   - The unit vectors ($\vec{n}_r$, $\vec{n}_{fr}$, $\vec{n}_e$, $\vec{n}_{fe}$) defining the orientation of the cutting edge.
   - The static chip thickness.

     $$t_c(\theta) = s_t \sin(\theta) \cos(\lambda) \cdot (\kappa \cdot \vec{b}) \tag{3.41}$$

   - The differential area, δA_r, and cutting edge length, dS.

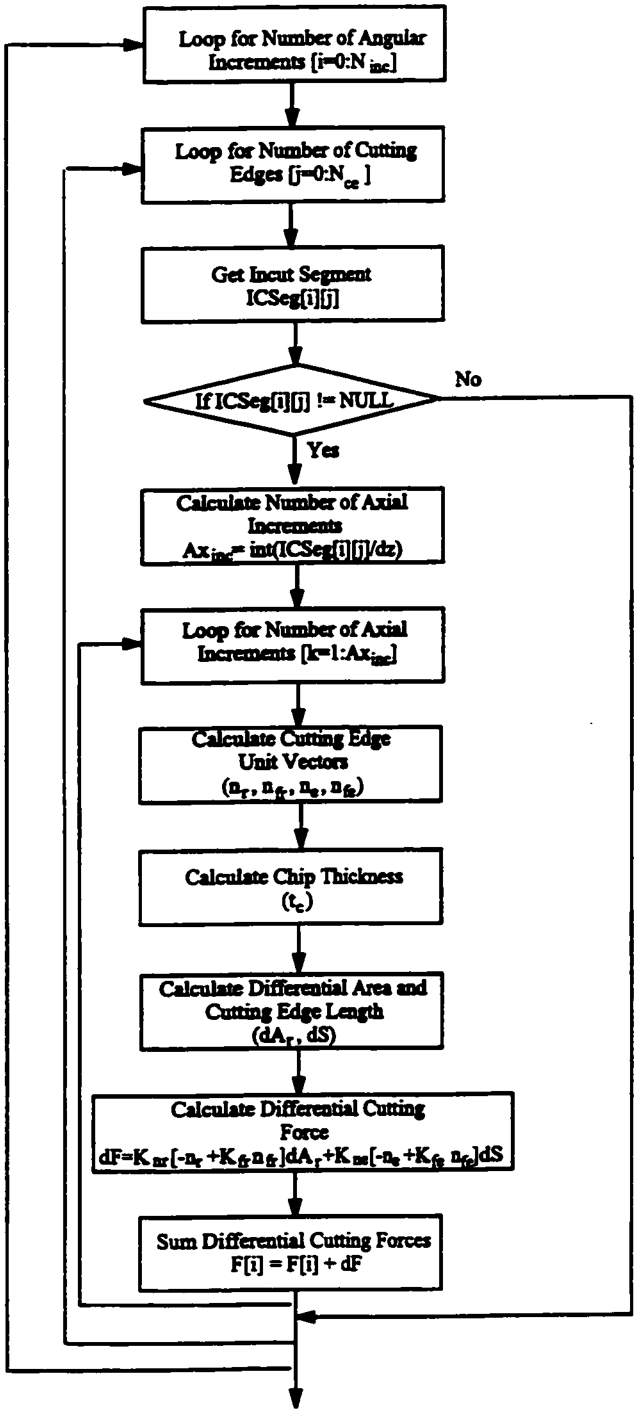


Figure 3.11: Static Force prediction Flow Chart

- The differential cutting force, dF

$$\begin{aligned} d\Im &= K_{nr} \cdot [-\vec{n}_r + K_{fr} \cdot \vec{n}_{fr}] \cdot \delta A_r \\ &\quad + K_{ne} \cdot [-\vec{n}_e + K_{fe} \cdot \vec{n}_{fe}] \cdot \delta S \end{aligned} \tag{3.42}$$

The total cutting force is calculated by summing the differential force for all axial elements.

$$\Im = \sum_{j=0}^{j \leq Ax_{inc}} d\Im[j] \tag{3.43}$$

3.7 Dynamic Simulation Methodology based on Control Points

This section describes the implementation of a strategy for dynamic modeling similar to those presented in [3, 39]. For complex sculptured machining operations the regenerative force, dynamic deflection model has been implemented as a piece-wise continuous solution. The solution is considered piece-wise continuous in the sense that the dynamic simulation is carried out at user specified intervals along the tool path (control points). At each control point on the tool path, the following steps are performed:

- The tool immersion (incut segments) is determined from the static tool/workpiece model.
- The tool immersion is used as integration limits for the regenerative force, dynamic deflection model.

- The dynamic simulation is performed for a user specified number of tool rotations.

The implementation of this strategy closely parallels the methodology presented in section 3.6. The implementation steps shown in figure 3.10 are used to define the cutting edge incut segments. However, the static force prediction block (figure 3.11) is substituted with the regenerative force, dynamic deflection model shown in figure 3.12. The cutting force and tool deflection are calculated for a number of user specified tool rotations $(0 : N_{rot})$. For each angular increment, i, and each cutting edge, j, the incut segment, $ICSeg[i][j]$ (determined from the static tool/workpiece model), is used as the integration limits for the force prediciton. For each incut segment the cutting edge is discretized into a number of axial elements, Ax_{inc} (equation 3.40). For each axial element the following are determined:

- The unit vectors ($\vec{n}_r$, $\vec{n}_{fr}$, $\vec{n}_e$, $\vec{n}_{fe}$) defining the orientation of the cutting edge.
- The chip thickness.

  $$t_c = s_{t,st} + DZ_0 - min(DZ_1, DZ_2, DZ_3) \tag{3.44}$$

  where $s_{t,st}$ is the static chip thickness given by equation 3.41, DZ_0 is the current deflection of the tool, and DZ_1, DZ_2, DZ_3 are the deflections of the previous 3 tooth periods.
- The differential area, δA_r, and cutting edge length, dS.

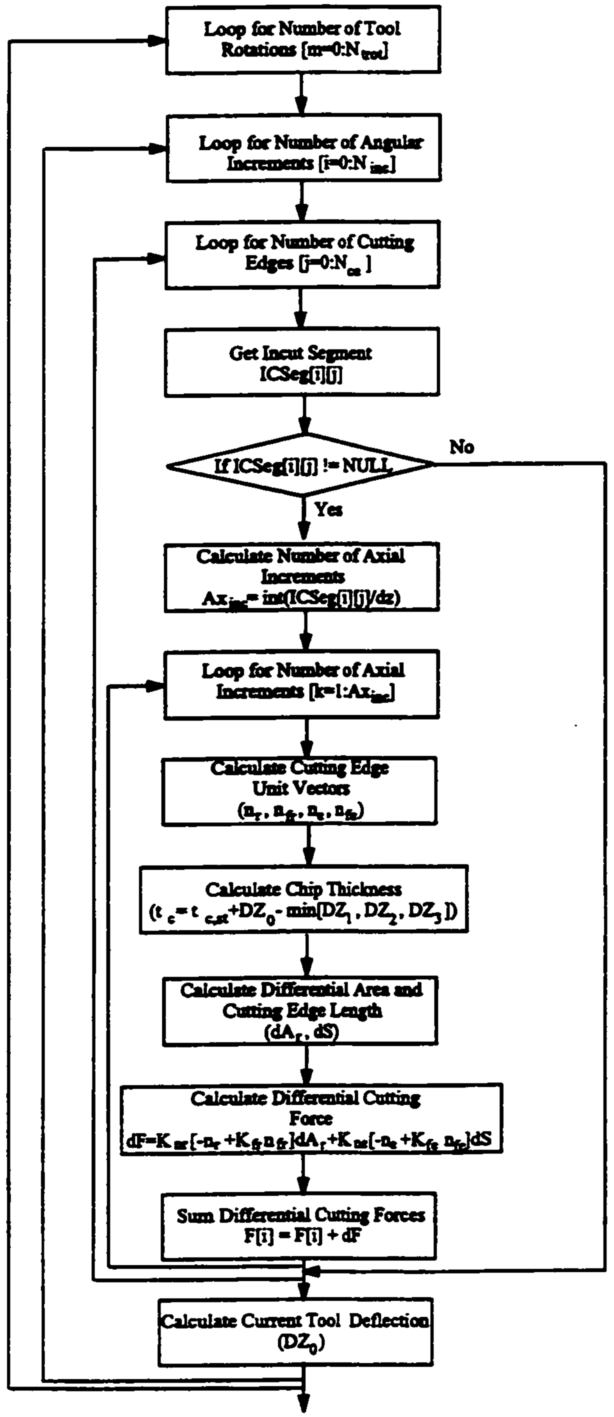


Figure 3.12: Dynamic Modeling Force prediction (Control Point)

- The differential cutting force, dF

$$\begin{aligned} d\Im &= K_{nr} \cdot [-\vec{n}_r + K_{fr} \cdot \vec{n}_{fr}] \cdot \delta A_r \\ &\quad + K_{ne} \cdot [-\vec{n}_e + K_{fe} \cdot \vec{n}_{fe}] \cdot \delta S \end{aligned} \tag{3.45}$$

The total cutting force is calculated by summing the differential force for all axial elements.

$$\Im = \sum_{j=0}^{j \leq Ax_{inc}} d\Im[j] \tag{3.46}$$

The current tool deflection can be determined by integrating the governing dynamic force equation in each coordinate direction (X,Y). The governing dynamic force equation is:

$$\Im_i = m\ddot{x}_i + c\dot{x}_i + kx_i \tag{3.47}$$

Integration is approximated by using a fourth order Runge-Kutta method [70].

3.8 Continuous Dynamic Simulation Methodology

The objective of the work presented in this section is to provide an enhanced dynamic mechanistic model for multi-axis machining. This work builds upon the modeling methodologies presented in sections 3.6 and 3.7. The dynamic process simulation methodology will be presented as a continuous solution for complex sculptured surface machining.

A continuous dynamic process simulation methodology for multi-axis machining needs to be formulated to include the following:

1. The simulation step size is given by a minimum time period.
2. The instantaneous location of the cutting edge is determined from the original tool centre (defined by the current tool path location) and the current deflection of the tool.
3. The instantaneous chip thickness is determined from the feed/tooth and the deflection of the previous N tooth periods.
4. Incorporate the deflection effects of previous tool path segments.

The minimum time period is defined by the dominant system natural frequencies. The dominant system natural frequencies for a typical machining operation (assuming a rigid part) include the spindle and tool natural frequencies. These values are typically determined by experimental modal analysis. The minimum time period is defined by the user. The NURBS representation of the tool cutting edge allows item 2 to be easily implemented. The instantaneous location of the cutting edge can be obtained by applying the translations and rotations for the tool centre and current deflection to the NURBS curve. To determine the instantaneous chip thickness (item 3), the deflection history of the tool needs to be retained. To incorporate the deflection effects of previous tool path segments the deflection information along a tool path must be linked to the specific tool path segment. Each tool path segment is represented by a data structure. For the static modeling methodology presented in section 3.6 this data structure contains:

- tool path tangent vector, $\vec{\tau}$
- swept surface representation
- tool path bounding box (static)

To include the effects of deflection, the following information must be added to each tool path data structure:

- actual tool path curve
- dynamic bounding box

Figure 3.13 depicts the information contained within each tool path data structure. The actual tool path curve is determined from the corresponding tool deflection history for each tool path segment. The dynamic bounding box is determined from the actual tool path curve. The maximum tool deflections normal to the tool path are used to calculate the size of the bounding box.

The implementation of the methodology for a continuous dynamic process simulation for multi axis machining of complex sculptured surfaces is shown in figure 3.14. The simulation begins by reading the tool path information from a CLData file. For each tool path segment a data structure is created. The static bounding box shown in figure 3.13 is used to initialize this data structure. The simulation proceeds by looping through each tool path segment. For each tool path segment the number of simulation time steps, n_{step}, is determined from the length of the tool path segment, TP_{len}, the feed per revolution, f_{rev}, and the number of simulation steps per revolution, N_{rev}.

$$n_{step} = \frac{TP_{len} \cdot N_{rev}}{f_{rev}} \tag{3.48}$$

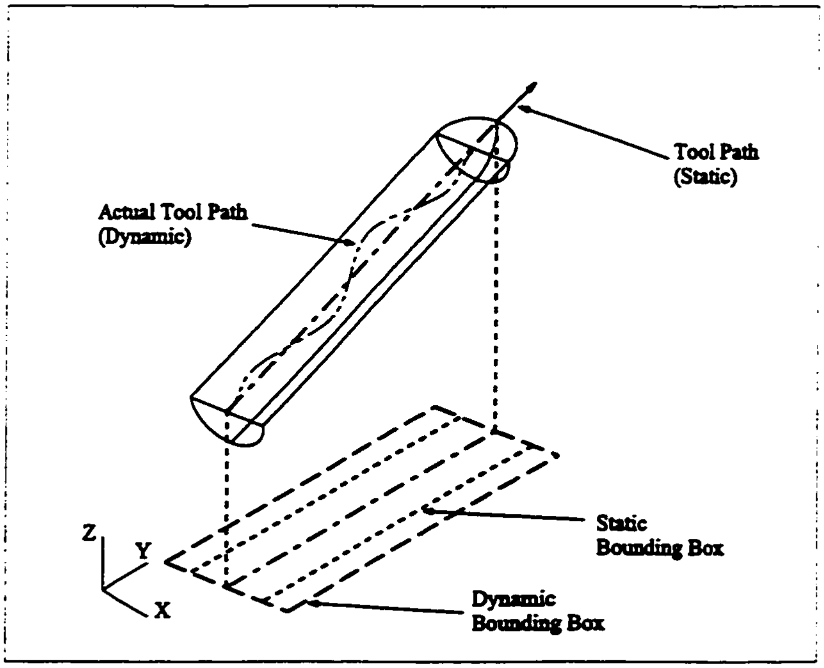


Figure 3.13: Dynamic Tool Path Information

For each simulation time step the following are performed:

1. The next point along the tool path, P_n, is determined. P_n is calculated by applying two translations (T_s,T_d) to the last point, P_o.

$$P_n = P_o + T_s + T_d \tag{3.49}$$

The translation are:

- T_s is a translation in a direction defined by the tool path tangent vector, with magnitude equal to the feed per time step

$$T_s = \frac{f_{rev}}{N_{rev}} \cdot \vec{\tau} \tag{3.50}$$

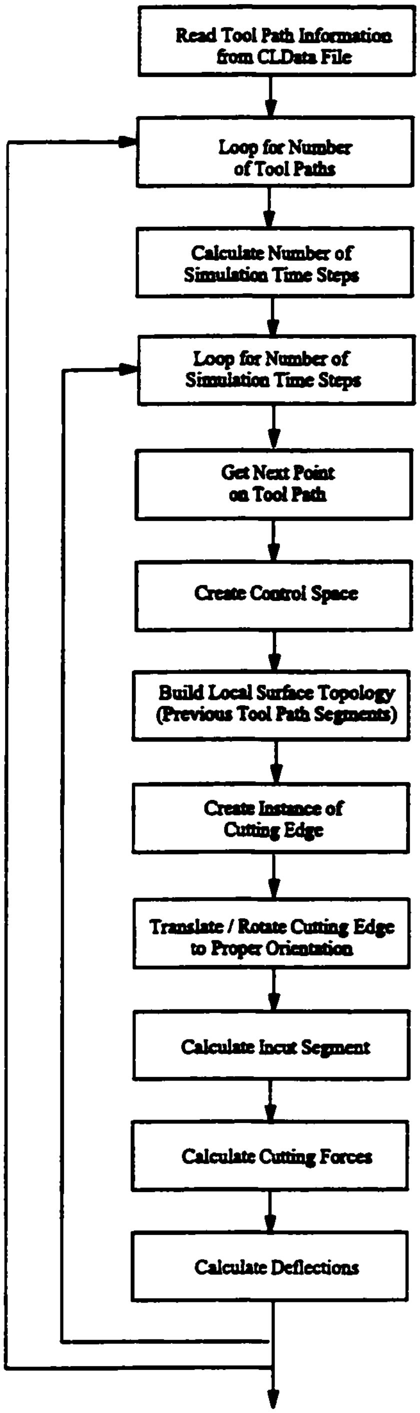


Figure 3.14: Continuous Dynamic Modeling Flow Chart

- T_d is a translation normal to the tool path, with magnitude equal to the current deflection, DZ_0

$$T_d = DZ_0 \cdot \vec{\tau}_n \tag{3.51}$$

2. A control space is defined for the current tool position.

3. The local surface topology is constructed for the current tool position. The local surface topology is defined by previous tool path segments that intersect the control space.

4. An instance of the tool cutting edge is created and translated to the current tool position, P_n. The tool cutting edge is then rotated about P_n by an angle, $d\theta = \frac{360}{N_{rev}}$

5. The cutting edge incut segment ($ICSeg$) is determined by classifying points on the cutting edge against the local surface topology.

6. The dynamic cutting force is calculated. Figure 3.15 shows a flow chart for the dynamic force prediction. For each cutting edge, j, the incut segment, $ICSeg[j]$, is used as the integration limits for the force prediciton. For each incut segment the cutting edge is discretized into a number of axial elements, Ax_{inc} (equation 3.40). For each axial element the following are determined:

    - The unit vectors ($\vec{n}_r$, $\vec{n}_{fr}$, $\vec{n}_e$, $\vec{n}_{fe}$) defining the orientation of the cutting edge.

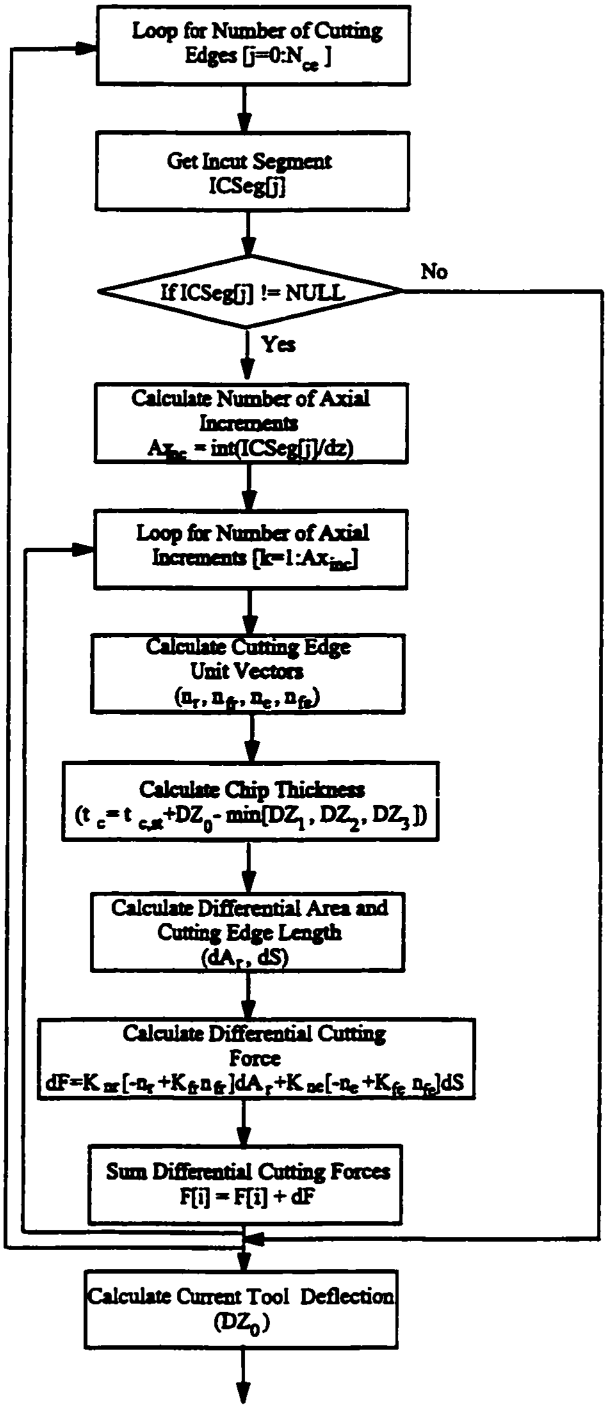


Figure 3.15: Dynamic Modeling Force prediction (Continuous)

- The chip thickness.

$$t_c = s_{t,st} + DZ_0 - min(DZ_1, DZ_2, DZ_3) \tag{3.52}$$

  where $s_{t,st}$ is the static chip thickness given by equation 3.41, DZ_0 is the current deflection of the tool, and DZ_1, DZ_2, DZ_3 are the deflections of the previous 3 tooth periods.

- The differential area, δA_r, and cutting edge length, dS.
- The differential cutting force, $d\Im$

$$\begin{aligned} d\Im &= K_{nr} \cdot [-\vec{n}_r + K_{fr} \cdot \vec{n}_{fr}] \cdot \delta A_r \\ &\quad + K_{ne} \cdot [-\vec{n}_e + K_{fe} \cdot \vec{n}_{fe}] \cdot \delta S \end{aligned} \tag{3.53}$$

The total cutting force is calculated by summing the differential force for all axial elements.

$$\Im = \sum_{j=0}^{j \leq Ax_{inc}} d\Im[j] \tag{3.54}$$

The current tool deflection can be determined by integrating the governing dynamic force equation in each coordinate direction (X,Y) . The governing dynamic force equation is:

$$\Im_i = m\ddot{x}_i + c\dot{x}_i + kx_i \tag{3.55}$$

Integration is approximated by using a fourth order Runge-Kutta method [70].

After each tool path segment has been simulated, the deflection history is utilized to set the size of the dynamic bounding box. The dynamic bounding box is then used in subsequent simulation steps to determine the local surface topology of the part.

3.9 Internet Based Machining System

The Internet is a global information system that provides a communications medium for connecting computers, and computing resources around the world. Since its early inception in the 1970's, the Internet has grown in size and popularity to provide many services and applications. The major Internet services and applications include: electronic mail, file transfer protocols, electronic discussion groups, search tools, and the world wide web, a hypertext based information medium. Commercial entities have also begun to leverage the Internet for various forms of e-commerce and/or e-business. In the area of engineering, the Internet offers a medium for technology integration. This involves both information exchange and providing access to engineering resource tools.

The Internet based machining system integrates advanced process simulation software, and state of the art multimedia and Internet technologies to realize an Internet enabled process optimization environment. This environment leads the manufacturing engineer through formalized procedures in which physics based process simulations allow process parameters and operational sequences to be evaluated and optimized with respect to desired product features. These features include dimension, surface quality, tool life, and cost and lead time reduction. The modeling methodologies presented in this chapter have been developed to accurate predict instantaneous chip load and cutting forces. The inputs for the simulation are the cutter location file and tool geometry. The outputs of the simulation are the instantaneous chip load, cutting forces, and tool deflection. The simulation outputs are utilized for optimizing process parameters through feed scheduling and changes to cutting conditions to reduce tool deflection. The process optimization provides an increase in the dimensional accuracy of the resulting part geometry, improves the quality of the surface produced, and increases tool life. All this occurs within an Internet enabled process optimization environment.

The Internet based machining system provides a formalized system for designing and proofing machining processes. The interactive nature of the Internet based implementation of the multi-axis machining simulation offers the user valuable insight into the machining process. This information can be used for reducing or eliminating the need for expensive process trials. Specific process problems can also be diagnosed more efficiently when this information is available. Having this information allows the user to optimize the machining process by testing different process parameters (feed, speed, tool paths, etc.). Process develoment can, therefore, occur iteratively. Implementing these development tools using Internet technologies has several advantages. It allows all of the development tools to be in a centralized location, allowing for quick and reliable upgrade paths. Internet accessibility also gives quick and easy access to these development tools to researchers and manufacturing engineers around the world.

The implementation of the Internet based machining system is based on a client/server model. The client, running on the user's web browser, makes requests for services. The server, running on the host computer, services the requests. The client and server are on different machines across the Internet. Communication between the client and server uses the TCP/IP (Internet Protocol). The implementation methodology and details on running the simulation are given in appendix E.

3.10 Summary

This chapter presented the modeling methodologies required for a generic simulation approach for multi-axis machining of complex sculptured surface parts.

A generalized approach is developed for representing an arbitrary cutting edge design, and the local surface topology of a complex sculptured surface part. A NURBS curve is used to represent the cutting edge profile. This approach offers the advantages of representing an arbitrary cutting edge design in a generic way, as well as providing standardized techniques for manipulating the location and orientation of the cutting edge. The local surface topology of the part is defined as those surfaces generated by previous tool paths in the vicinity of the current tool position. The local surface topology of the part is represented without using a computationally expensive CAD system. A systematic prediction technique was then developed to determine the instantaneous tool/part interaction during machining. The methodology employed here determines cutting edge in-cut segments by determining the intersection between the NURBS curve representation of the cutting edge and the defined local surface topology.

A comprehensive cutting force model for multi-axis machining was presented. The cutting force model considers both normal and friction components of force on the rake and clearance faces on the cutting edge. A systematic model calibration procedure was developed to incorporate the effects of varying cutting edge geometry, cutting speeds, and feed rates.

This chapter also presents an enhanced dynamic mechanistic modeling methodology for multi-axis machining. The dynamic process simulation methodology is presented as a continuous solution for complex sculptured surface machining. The methodology is formulated to include the instantaneous deflection of the tool, and the tool deflection history over the entire tool path.

This chapter also presented the concept and implementation of an Internet based facility for multi-axis milling process simulation and optimization. The facility integrates advanced process simulation software, and state of the art multimedia and Internet technologies to realize an Internet enabled process optimization environment for machining systems.

Chapter 4

Feed Rate Scheduling

4.1 Introduction

During machining of complex sculptured surfaces, the chip geometry is constantly changing along the tool path. This causes wide fluctuations in the process variables (cutting force and tool deflection). In the worst case, the tool may break or cutting may become unstable (chatter) and/or the quality of the finished product may be unacceptable. To ensure stable machining conditions and guarantee the quality of the finished product, conservative process parameters are typically used (feeds and speeds). The result is lower productivity. Feed rate scheduling can be used to optimize the manufacture of complex sculptured surface parts. The productivity will be increased with guaranteed machining integrity. Section 4.2 develops the feed scheduling methodology for multi-axis machining.

The machine tool feed drive performance capabilities have not been integrated with feed scheduling in any of the reviewed literature. Integrating a model of the machine tool feed drive with a feed scheduling strategy is necessary to smooth the feed rates and to ensure that the scheduled feed rates and corresponding accelerations are within the machine performance limitations. Section 4.3 presents two strategies for integrating the feed drive performance limitations with feed scheduling (Acc/Dec time constant). The term Acc/Dec is used throughout the following sections to describe the different phases of acceleration that the machine experiences during changes in feed rate. The time that the acceleration rate is increasing is referred to as the Acc phase. The time that the acceleration rate is decreasing is referred to as the Dec phase. The first strategy implements feed scheduling considering the feed drive Acc/Dec time constant. The second strategy optimizes the use of the feed drive capabilities for feed scheduling.

A summary of the feed scheduling methodologies is given in section 4.4.

4.2 Methodology for Multi-Axis Machining

To properly implement feed rate scheduling, a set of process constraints must be formulated. Several authors have presented various process constraints for milling applications [39, 79, 8]. The cited authors included shank and tooth breakage constraints, and surface dimensional error constraints. In practice, the above process constraints can be satisfied by imposing either a maximum chip load constraint or maximum force constraint. Maximum chip load and force constraints have been implemented here for feed rate scheduling. The chosen process constraint is determined by the type of machining operation. For roughing machining operations the maximum force constraint is imposed. This will eliminate force spikes during cutting that could either chip or break the cutting edge. The maximum force constraint can also be used to limit tool deflections. For semi-finishing or finishing operations the maximum chipload constraint can be used to control surface finish and/or increase productivity.

Feed rate scheduling is performed at user specified intervals along the given tool path (control points). The number of control points chosen depends on the accuracy requirement and the available computing tool. The control point density can either vary or be uniform along the tool path, depending on the curvature of the surface [58]. For each control point, the cutting geometry and cutting forces are extracted using the modeling methodologies presented in chapter 3. The assumption here is that the machining process is stable (ie no chatter). The process dynamics are not incorporated with feed scheduling. The static cutting geometry and cutting force are used for feed scheduling. The maximum allowable feed rate is scheduled by applying the chosen process constraint (maximum chip load or maximum force).

The procedure implemented here is illustrated by the flow chart given in figure 4.1. The maximum force constraint is used to develop the methodology. The procedure is similar for feed scheduling using a maximum chipload constraint. The control points $P(k)$ are defined at equal distance increments ΔX along the tool path:

$$P(k) = P(k-1) + \Delta X \tag{4.1}$$

At each distance increment ΔX, an iterative technique is used to solve for the optimal feed rate. As a "first guess", the feed rate is set to the maximum allowable feed rate:

$$F_S(k) = F_{max} \tag{4.2}$$

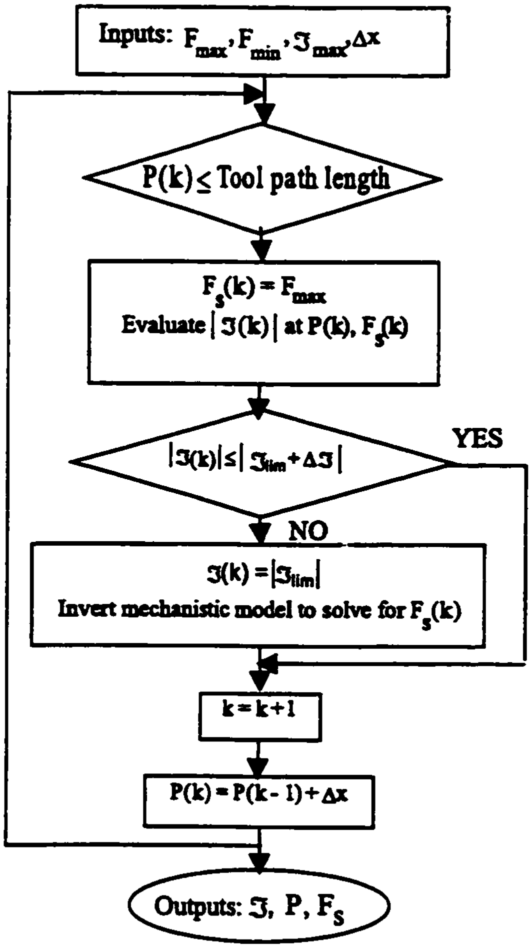


Figure 4.1: Flow chart for feed scheduling based on control point strategy

The instantaneous cut geometry is then extracted for the "first guess" of the feed rate value and the cutting force $\Im(k)$ is evaluated. If the cutting force value does not exceed the maximum allowable force,

$$|\Im(k)| \leq |\Im_{lim} + \delta\Im| \tag{4.3}$$

the tool move is incremented and the following control point is considered. The constraint given in equation (4.3) includes the machine condition when air cutting occurs. Air cutting can occur when approaching the workpiece, when leaving the workpiece or when there is a pocket in the workpiece. When the cutting force value exceeds the maximum allowable force,

$$|\Im(k)| > |\Im_{lim} + \delta\Im| \tag{4.4}$$

the feed rate value $F_S(k)$ should be decreased. The cutting force value is then set to the limit value of the cutting force:

$$\Im(k) = \Im_{lim} \tag{4.5}$$

The mechanistic model is then inverted to solve for the feed rate $F_S(k)$. The feed rate $F_S(k)$ is determined by applying a bisection method. The iterations continue until the predicted force is within a given tolerance of the target force value. The tool move is incremented and the next control point is considered.

This procedure is repeated for each tool move resulting in a set of feed rates for a given machining operation.

4.3 Incorporating Feed Drive Performance Capabilities

The feed scheduling methodology presented in section 4.2 was formulated independently of the machine tool feed drive performance capabilities. The assumption made by this methodology is that the scheduled commanded feed rate, F_S, is always equal to the actual feed rate, F_a, that the machine tool achieves during machining. This would mean that the step changes in feed rate specified by the feed scheduler are achievable instantaneously by the machine tool feed drives. Practically, this is not the case. Each feed drive on a machine tool has a set Acc/Dec profile that it must follow when a step change of feed rate is specified. Therefore, the scheduled commanded feed rate, F_S, is not equal to the actual feed rate, F_a, during Acc/Dec phases. There are several implication for feed scheduling. The actual feed rates attained by the machine tool during Acc/Dec may cause the process constraints (force, chipload) to be violated. The force prediction is also not reliable during the Acc/Dec phases. If the feed rate step between successive control points is large, the time interval between control points will be small. If the time interval is less than the Acc/Dec time for the feed drive, the actual feed will be truncated so that the machine tool tracks its path. This may cause the process constraints to be violated.

It is evident from the above discussion that the performance capabilities of the machine tool feed drive need to be integrated with the feed scheduling methodology. The first step is to develop a model of the feed drive dynamics that can be used to generate the Acc/Dec profile. The second step is to formulate a feed scheduling methodology that incorporates the feed drive Acc/Dec profile.

The first step is to identify the low frequency dynamics of the feed drive system. The identification is limited to the transfer function relating the actual feed rate, assumed to be proportional to the motor shaft velocity, to the desired feed rate specified in the CLData file. An empirical method based on experimental identification has been adopted since it results in a more accurate model for the real feed drive dynamics. A systematic procedure of identification based on time series modeling is developed. In this procedure, specific criteria are introduced to evaluate the adequacy of the model and, consequently, to select the appropriate model structure and orders. The input excitation used for identification is considered to account for both the spectrum and the shape. The input excitation is specified as a multilevel sequence of feed rate with mainly low frequency components. The input is shaped so that the identified model is valid over a wide range of the feed rate within the physical limits of the machine. The identified model is an average of the forward and backward models of the feed drive system. The application of the identification procedure results in an ARMAX model with the following orders: $n_a = 3$, $n_b = 1$, $n_c = 1$, and $n_k = 1$. This model is transformed into a FIR filter and used as an Acc/Dec processor to generate the actual feed rate from the desired feed rate specified by the CLData file. The transformation considers variable, user-specified sampling frequencies different from that used for identification. The FIR filter coefficients have been normalized so that the actual feed rate converges to the desired feed rate at a steady state. The application of the Acc/Dec processor only describes the acceleration or the deceleration of the feed drive. It does not account for the effects of friction. A detailed overview of the identification procedure is given in appendix D.

The second step is to formulate a feed scheduling methodology that incorporates the feed drive Acc/Dec profile. The following sections detail the strategies that are developed and implemented.

4.3.1 Acc/Dec Time Constant Strategy

In order to achieve the desired feed rate at the appropriate position, it is necessary to take into account the dynamic capabilities of the feed drive system. A time domain based strategy for feed scheduling is introduced in this section. This strategy considers the feed drive dynamics described by the Acc/Dec profile. The objective of this strategy is to guarantee that the actual feed rate, F_a, of the machine will reach the commanded feed rate, F_S. In this strategy, the commanded feed rate, F_S, the actual feed rate, F_a, and the cutting force are simultaneously generated for a given machining operation. This procedure is illustrated by the flow chart given in figure 4.2.

Given that the feed drive is moving initially with a constant feed rate (acceleration equal to zero), a step change in the feed rate can only be achieved in a time period equal to the machine Acc/Dec time constant, τ_{total}. To guarantee that the actual feed rate, F_a, reaches the commanded feed rate, F_S, the tool move is split into equal time segments. The period of each time segment is equal to the Acc/Dec time constant, τ_{total}. Each time segment will correspond to one block of tool motion. For each block of motion, the commanded feed rate value should be determined so that the actual feed rate is guaranteed to respect the force constraint during the acceleration or deceleration phase. A sampling period is introduced to subdivide the time segment, τ_{total}, into smaller periods of time. This period is specified equal to the tooth passing

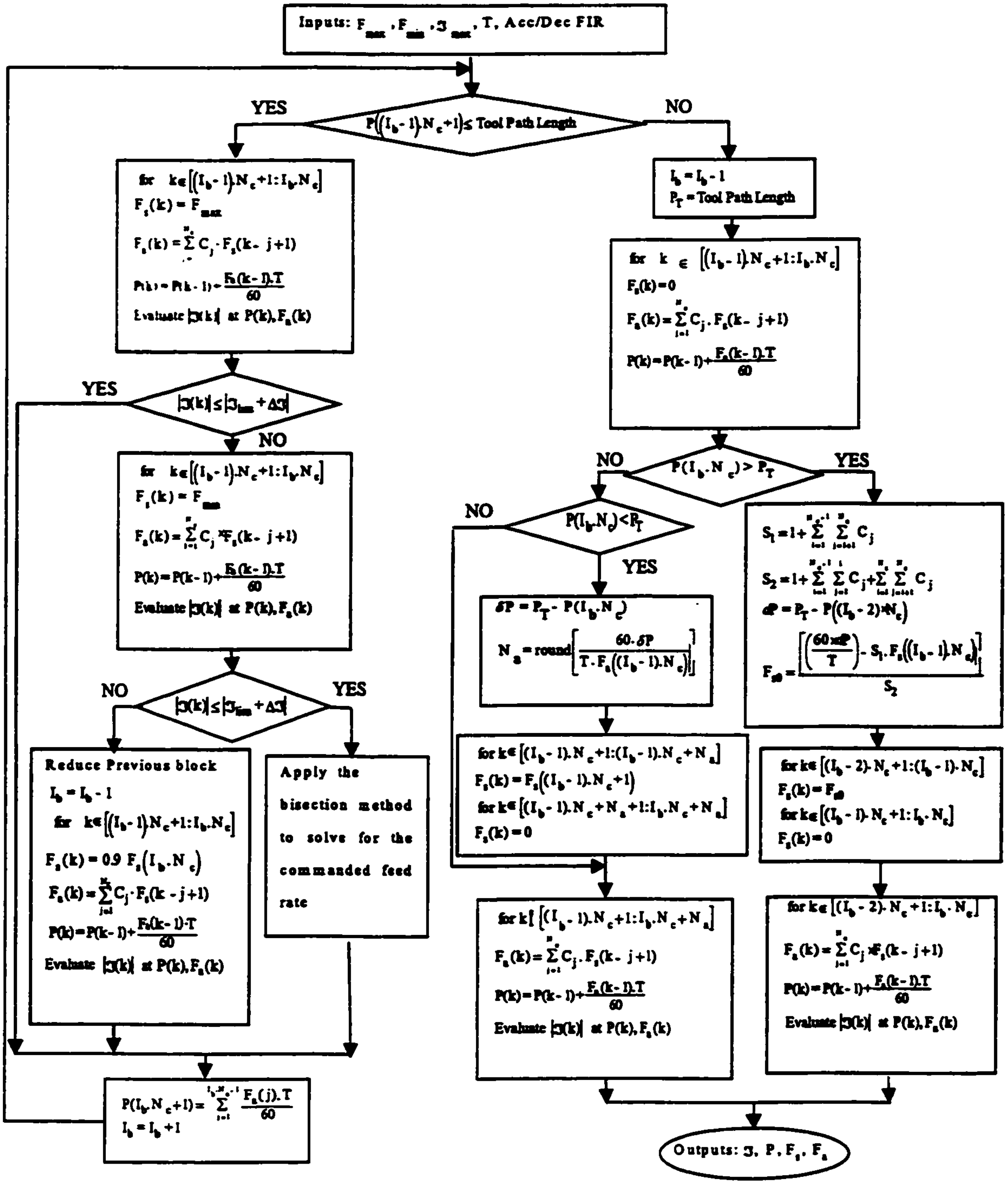


Figure 4.2: Flow chart for feed scheduling based on Acc/Dec time constant

period:

$$T = \frac{60}{N_{th} \cdot N_{rpm}} \tag{4.6}$$

where N_{th} is the number of teeth on the cutter, and N_{rpm} is the spindle rpm. Using this sampling period, the Acc/Dec profile is transformed into a FIR filter (see appendix D). Each block of motion will have a number of periods equal to the number of coefficients of the FIR filter, N_c.

The distance traveled within each block is dependent on the commanded feed rate, and can not be known a priori. An iterative strategy is proposed to determine the incremental tool motion. The following paragraphs provide an overview of the feed scheduling strategy. The first paragraph describes the generation of the intermediate tool motion blocks. The second paragraph describes the generation of the tool motion blocks at the end of the tool path.

Generation of intermediate tool motion blocks

For each block, I_b, the commanded feed rate is set to the maximum allowable feed rate value:

$$F_S(k) = F_{max} \; for \; k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c] \tag{4.7}$$

Then, the actual feed rate is evaluated using the coefficients, C_j, of the FIR:

$$F_a(k) = \sum_{j=1}^{N_c} C_j \cdot F_S(k - j + 1) \; for \; k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c] \tag{4.8}$$

Based on the actual feed rate, the tool moves, $P(k)$, can be evaluated:

$$P(k) = P_0 + \sum_{i=2}^{k} F_a(i-1) \cdot \frac{T}{60} \; for \; k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c] \quad (4.9)$$

where P_0 is the initial position of the tool. The instantaneous cut geometry is then extracted at each tool position, $P(k)$, and the cutting force, $\Im(k)$, is evaluated.

If the cutting force value does not exceed the target force for all the increments k, $k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c]$, the number of the tool motion blocks is incremented and the next block is considered.

If the cutting force value, $\Im(k)$, exceeds the target force for a given increment, k, the value of the actual feed rate, $F_a(k)$, should be decreased. To decrease this value, the commanded feed rate for this block should be reduced. A bisection method is used to solve for the commanded feed rate value. The upper bound for the bisection method is set as the maximum allowable feed rate. This upper bound has been treated above. The lower bound is set to the minimum allowable feed rate. The lower bound is first checked to ensure its validity. The commanded feed rate is set equal to the minimum allowable feed rate.

$$F_S(k) = F_{min} \; for \; k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c] \quad (4.10)$$

The actual feed rate, $F_a(k)$, the tool position, $P(k)$, and the cutting force, $\Im(k)$, are evaluated. If all the cutting forces, $\Im(k)$, are less than the target force, $|\Im(k)| \leq |\Im_{lim} + \delta\Im|$, the lower bound is valid. The bisection method can proceed to solve for the optimum commanded feed rate. The iterations continue until the change in the commanded feed rate is within a specified tolerance.

If any of the cutting forces, $\Im(k)$, are greater than the target force, $|\Im(k)| > |\Im_{lim} + \delta\Im|$, the lower bound is not valid. The commanded feed rate of the previous tool motion block should then be reduced. The number of tool motion blocks is decremented, $I_b = I_b - 1$, and the commanded feed rate is reduced by a user-defined coefficient:

$$F_S(k) \;=\; 0.9F_S(I_b \cdot N_c) \; for \; k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c] \tag{4.11}$$

The actual feed rate and the cutting force are re-evaluated for this block. The cutting forces evaluated for this block will always be less than the target force because the actual feed rates have been reduced.

The tool position at the starting of the next block is expressed as:

$$P(I_b \cdot N_c + 1) \;=\; P_0 + \sum_{i=2}^{I_b N_c + 1} F_a(i-1) \cdot \frac{T}{60} \tag{4.12}$$

While this position does not exceed the tool path length, the number of the tool motion blocks is incremented and the above procedure is repeated. When the starting point of the next block exceeds the tool path length, the generation of the tool motion blocks is adapted to decelerate the feed drive to stop at the end of the tool path. This will be studied in the next section.

Generation of the tool motion blocks at the end of the tool path

When the starting point of the current block, I_b, of the tool motion exceeds the tool path length, the number of the blocks should be decremented, $I_b = I_b - 1$. This block is adapted to allow the feed drive to decelerate to zero velocity at the end of the path.

The target tool position at the end of the path is designated by P_T. To decelerate to zero velocity, the machine requires an interval of time equal to the Acc/Dec time constant, τ_{total}. The number of time increments needed for this deceleration is equal to the number of FIR filter coefficients. The commanded feed rate is set to zero for the deceleration block:

$$F_S(k) \;=\; 0 \; for \; k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c] \tag{4.13}$$

The actual feed rate is evaluated using the Acc/Dec FIR filter coefficients. The tool position is consequently determined for each increment of time k. The distance traveled by the tool can be equal to the target tool position, $P(I_b \cdot N_c) = P_T$, overshoot the target tool position, $P(I_b \cdot N_c) > P_T$, or undershoot the target tool position, $P(I_b \cdot N_c) < P_T$. If the distance traveled equals the target tool position, the cutting force is evaluated and the generation of the tool motion blocks is completed. Special procedures are necessary for other cases.

Overshoot target tool position If the distance traveled by the tool exceeds the target tool position, $P(I_b \cdot N_c) > P_T$, the commanded feed rate of the previous block should be reduced. Given the tool position at the end of the second last block, $P\left((I_b - 2) \cdot N_c\right)$, and the target tool position, P_T, the distance the tool should travel within two blocks is equal to δP:

$$\delta P \;=\; P_T - P\left((I_b - 2) \cdot N_c\right) \tag{4.14}$$

The commanded feed rate for the last block is equal to zero to allow the deceleration of the feed drive to stop at the end of the path. Suppose that the feed rate for the second last block is equal to F_{S0}, then:

$$F_S(k) \;=\; F_{S0}\; for\; k \in [(I_b - 2)\cdot N_c + 1 : (I_b - 1)\cdot N_c] \tag{4.15}$$

Using the FIR filter coefficients to express the actual feed rate for each increment of time k, $k \in [(I_b - 2)\cdot N_c + 1 : I_b \cdot N_c]$ and using equations (4.13), (4.14), and (4.15), the unknown is determined by the following expression:

$$F_{S0} \;=\; \left[\frac{60\delta P}{T} - S_1 \cdot F_S\left((I_b - 1)\cdot N_c\right)\right]\frac{1}{S_2} \tag{4.16}$$

where $S_1 = 1 + \sum_{i=1}^{N_c-1}\sum_{j=i+1}^{N_c} C_j$ and $S_2 = 1 + \sum_{i=1}^{N_c-1}\sum_{j=1}^{i} C_j + \sum_{i=1}^{N_c}\sum_{j=i+1}^{N_c} C_j$. Once the commanded feed rate is determined, the actual feed rate, $F_a(k)$, the tool position, $P(k)$, and the cutting force, $\Im(k)$, are evaluated for each increment of time k, $k \in [(I_b - 2)\cdot N_c + 1 : I_b \cdot N_c]$. The generation of the tool motion blocks is completed.

Undershoot target tool position If the distance traveled by the tool undershoots the target tool position, $P(I_b \cdot N_c) < P_T$, the tool must travel the remaining path distance δP with a constant feed rate. The constant feed rate is equal to the actual feed rate at the end of the last block, $F_a\left((I_b - 1)\cdot N_c\right)$. The remaining path distance δP is equal to:

$$\delta P \;=\; P_T - P\left(I_b \cdot N_c\right) \tag{4.17}$$

To travel the distance δP at the constant feed rate, a number of time increments are added. This number, designated N_a, is estimated by the formula:

$$N_a = round\left(\frac{60\delta P}{T \cdot F_a\left((I_b - 1) \cdot N_c\right)}\right) \tag{4.18}$$

The commanded feed rate and the actual feed rate are adjusted accordingly. The tool position and the cutting force are then evaluated for each increment of time k, $k \in [(I_b - 1) \cdot N_c + 1 : I_b \cdot N_c + N_a]$. The generation of the tool motion blocks is then complete.

4.3.2 Optimized Strategy

This strategy is formulated to achieve higher productivity and to improve the machining integrity. This strategy combines different constraints and various criteria in modifying the feed rate to maintain a constant cutting force. The constraints include the cutting force magnitude, the feed rate boundaries, and the characteristics of the Acc/Dec profile of the feed drive system. The criteria are the maximum production rate, and guaranteeing the machining integrity and safety. To generate the tool motion, the optimized feed scheduling strategy uses search and decision logic. This strategy is very complex due to large number of sequences in the logic. Flow charts are given to indicate clearly the logic sequences.

Problem formulation

Suppose that the tool is moving along a straight line from a starting position, P_0, to an end position, P_e. At both positions, P_0 and P_e, the feed rate and the acceleration of the feed drive are zero. The feed drive system has to accelerate at the start of the path to begin the tool motion. At the end of the path the tool motion is decelerated to reach the position P_e with zero velocity and zero acceleration. To generate the tool motion, the optimized feed scheduling strategy aims to generate the commanded feed rate, F_S, so that:

- The commanded feed rate, F_S, is within the limits of the feed drive system. The upper limit is set equal to the maximum allowable feed rate. The feed drive has a lower limit for the feed rate when there is no reverse direction in the motion.
- To stop tool motion at the end of the path, the feed drive system needs an interval of time equal to the Acc/Dec time constant, τ_{total}. This time interval allows deceleration from any given feed rate to zero velocity. The commanded feed rate, F_S, should be set equal to zero for the N_c last lags. N_c is the number of the FIR coefficients.
- The actual feed rate, F_a, respects the Acc/Dec capabilities of the feed drive system. The relationship between the commanded feed rate and the actual feed rate is described by the Acc/Dec FIR given in appendix D.
- The actual feed rate, F_a, guarantees that the cutting force, $\Im$, respects the target force constraint, $\Im_{lim}$. The variation of the cut geometry can result in a rapid change in the cutting force. When this change is composed of a rapid decrease followed by a rapid increase in the force magnitude, the feed drive has to accelerate to reach a higher feed rate, then decelerate to reach a lower value of the feed rate. Due to the Acc/Dec capabilities of the feed drive, the high feed rate may not be attained. The cutting force magnitude will be lower than

the target force in this case. The target force constraint is, therefore, defined so that the cutting force is less than or equal to the target force value. The difference $(\Im_{lim} - \Im(k))$ should also be minimized.

- The tool motion is determined to maximize the production rate.

For a given time period, the above problem can be formulated as:

$$find\,a\,vector\ F_S\ of\ a\,size\ N\ so\,that:$$

$$\left\{\begin{array}{l} \left\{\begin{array}{ll} F_{min} \le F_S(k) \le F_{max} & for\ k \in [1 : N - N_c] \\ F_S(k) = 0 & for\ k \in [N - N_c + 1 : N] \end{array}\right. \\ \qquad \left\{\begin{array}{l} for\ k \in [1 : N] \\ F_a(k) = \sum_{j=1}^{N_c} C_j \cdot F_S(k - j + 1) \\ |\Im(k)| \le |\Im_{lim}| \\ min\,(\Im_{lim} - \Im(k)) \\ min(N) \end{array}\right. \end{array}\right. \tag{4.19}$$

This problem is highly nonlinear. An iterative solution to generate the tool motion is proposed. The above problem (4.19) is decomposed into two parts. In the first part, the time is incremented gradually, $k = k + 1$, and the tool motion along the tool path is generated as if there is no stop at the end of the path. This is equivalent to the problem (4.19) where the equation $F_S(k) = 0 \;\; for\ k \in [N - N_c + 1 : N]$ is removed:

$$find\ a\ vector\ F_S\ of\ a\ size\ N\ so\ that:$$

$$\begin{cases} F_{min} \leq F_S(k) \leq F_{max} \;\; for\ k \in [1 : N - N_c] \\ \begin{cases} for\ k \in [1 : N] \\ F_a(k) = \sum_{j=1}^{N_c} C_j \cdot F_S(k - j + 1) \\ |\Im(k)| \leq |\Im_{lim}| \\ min\,(\Im_{lim} - \Im(k)) \\ min(N) \end{cases} \end{cases} \tag{4.20}$$

The second part of the solution is to adjust the commanded feed rate generated from the first part to allow the feed drive to stop at the end of the path. For this part, the formulation of the problem given by the system of equations (4.19) holds.

These two parts will be described in the following sections; generation of the tool motion along the tool path, and generation of the feed drive deceleration at the end of the tool path respectively. A flow chart for the generation of the tool motion is depicted in figure 4.3.

Generation of the tool motion along the tool path

A sampling period is introduced to generate the tool motion increments. The sampling period is specified equal to the tooth passing period:

$$T = \frac{60}{N_{th} \cdot N_{rpm}} \tag{4.21}$$

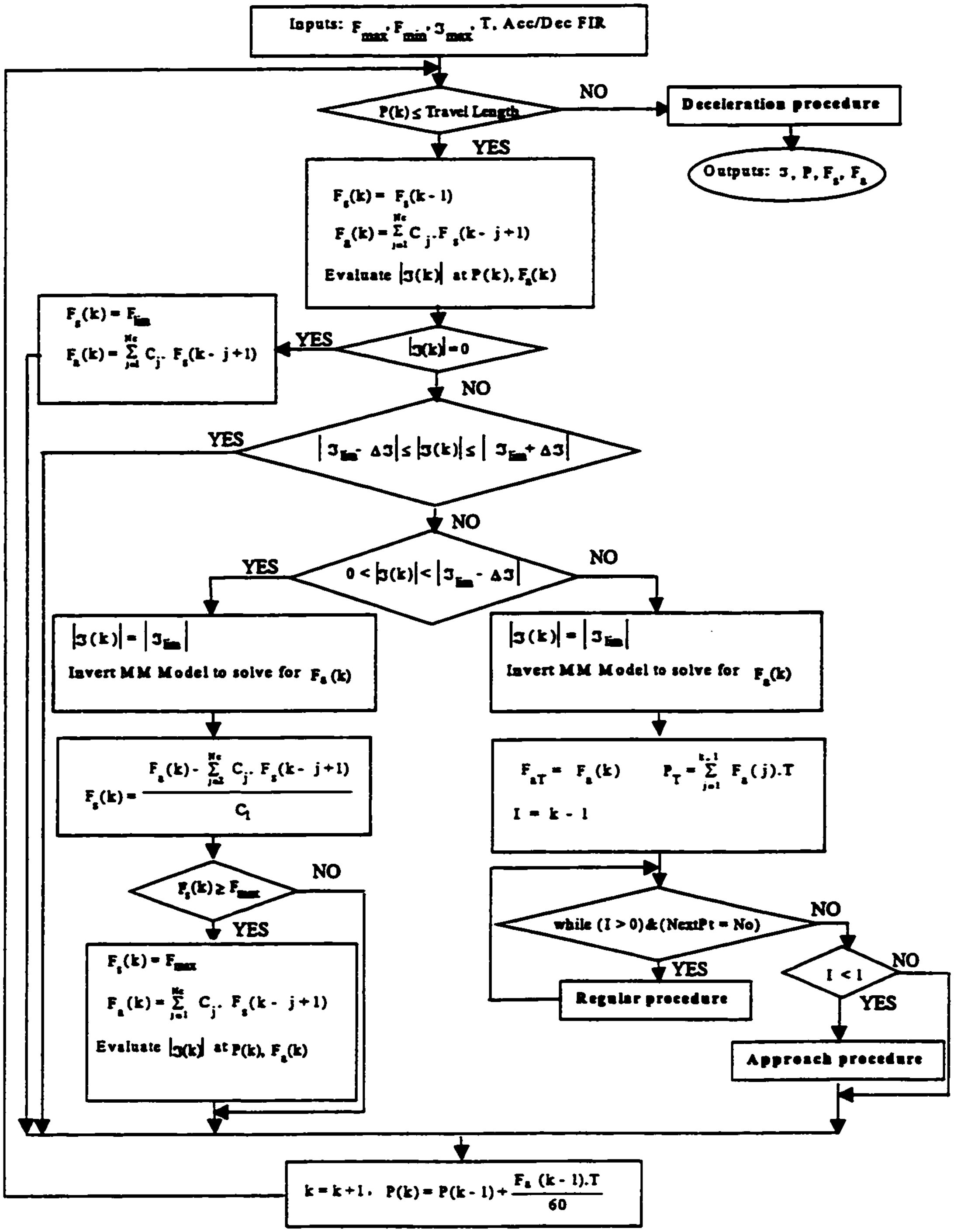


Figure 4.3: Flow chart for the "generation of the tool motion"

where N_{th} is the number of teeth on the cutter, and N_{rpm} is the spindle rpm. Using this sampling period, the Acc/Dec profile is transformed into a FIR filter (see appendix D).

Along the tool path, the cutting force is evaluated for each sample period, $t = k \cdot T$. The tool position, $P(k)$, and the actual feed rate, $F_a(k)$, are determined to estimate the in-cut geometry. The tool position, $P(k)$, is determined given the tool position and the actual feed rate for the previous lag:

$$P(k) = P(k-1) + F_a(k-1) \cdot \frac{T}{60} \tag{4.22}$$

The actual feed rate, $F_a(k)$, is determined by a first guess of the current commanded feed rate. The commanded feed rate, $F_S(k)$, is set equal to the previous commanded feed rate, $F_S(k-1)$:

$$F_S(k) = F_S(k-1) \tag{4.23}$$

This first guess is valid because the commanded feed rate of the previous lag is within the limits of the feed drive system. The feed rate, $F_a(k)$, is evaluated based on the Acc/Dec profile:

$$F_a(k) = \sum_{j=1}^{N_c} C_j \cdot F_S(k-j+1) \tag{4.24}$$

The mechanistic model is invoked to evaluate the magnitude of the resultant cutting force, $|\Im(k)|$. Given a tolerance interval, $\delta\Im$, on the target force value, four cases are identified:

- The force magnitude is null (air cutting) $|\Im(k)| = 0$,
- The force magnitude is within the tolerance interval of the target force, $|\Im_{lim} - \delta\Im| \leq |\Im(k)| \leq |\Im_{lim} + \delta\Im|$,
- The force magnitude is less than the maximum value of the target force and not null, $0 < |\Im(k)| \leq |\Im_{lim} - \delta\Im|$,
- The force magnitude is greater than the maximum value, $|\Im_{lim} - \delta\Im| < |\Im(k)|$.

Case 1 $|\Im(k)| = 0$ When air cutting, the actual feed rate $F_a(k)$ should be increased as fast as possible. The maximum value that the actual feed rate can reach within a time period is obtained when the commanded feed rate, $F_S(k)$, is set to the upper limit of the feed rate, F_{max}.

$$F_S(k) = F_{max} \tag{4.25}$$

The actual feed rate, $F_a(k)$, is evaluated based on the Acc/Dec profile:

$$F_a(k) = \sum_{j=1}^{N_c} C_j \cdot F_S(k-j+1) \tag{4.26}$$

The time is incremented, $k = k+1$, and the next tool position, $P(k+1)$, is considered.

Case 2 $|\Im_{lim} - \delta\Im| \leq |\Im(k)| \leq |\Im_{lim} + \delta\Im|$ The solution is accepted. The time is incremented, $k = k+1$, and the next tool position, $P(k+1)$, is considered.

Case 3 $0 < |\Im(k)| \leq |\Im_{lim} - \delta\Im|$ The cutting force magnitude is less than the target force but it is not null. To minimize the difference between the target force, $|\Im_{lim}|$, and the actual cutting force, $|\Im(k)|$, $min\,(\Im_{lim} - \Im(k))$, the actual feed rate, $F_a(k)$, should be increased. The optimal value of the current actual feed rate is obtained when the cutting force magnitude, $|\Im(k)|$, is equal to the target force value, $|\Im_{lim}|$:

$$|\Im(k)| = |\Im_{lim}| \tag{4.27}$$

The mechanistic force model is inverted to determine the actual feed rate, $F_a(k)$, that satisfies equation (4.27). The feed rate, $F_a(k)$, is determined using a bisection method. The iterations continue until the actual cutting force is within a given tolerance of the target force value. Due to the Acc/Dec profile of the feed drive system, the resulting actual feed rate, $F_a(k)$, may not be attainable if the commanded feed rate, $F_S(k)$, exceeds the limits of the machine. The commanded feed rate, $F_S(k)$, is evaluated using the Acc/Dec FIR filter definition:

$$F_S(k) = \frac{F_a(k) - \sum_{j=2}^{N_c} C_j \cdot F_S(k-j+1)}{C_1} \tag{4.28}$$

In this case, the objective is to increase the actual feed rate, $F_a(k)$. The only violation that can occur is that the commanded feed rate, $F_S(k)$, exceeds the maximum value of the feed rate, $F_S(k) > F_{max}$.

If the commanded feed rate, $F_S(k)$, does not exceed the maximum value of the feed rate, $F_S(k) \leq F_{max}$, the time is incremented, $k = k + 1$, and the next tool position, $P(k+1)$, is considered.

If the commanded feed rate, $F_S(k)$, exceeds the maximum value of the feed rate, $F_S(k) > F_{max}$, the commanded feed rate, $F_S(k)$, is set equal to the maximum value of the feed rate:

$$F_S(k) \quad = \quad F_{max} \tag{4.29}$$

The actual feed rate, $F_a(k)$, is then re-evaluated based on the Acc/Dec FIR filter. The mechanistic force model is invoked to evaluate the magnitude of the cutting force. $\Im(k)$. The resulting cutting force is less than the target force. This specific case points out how the feed drive system capabilities do not allow a change in the feed rate to occur as fast as changes in the cutting geometry. Maintaining a constant cutting force is not possible in this case. The actual feed rate value, $F_a(k)$, is the best that the feed drive can achieve. The time is then incremented, $k = k + 1$, and the next tool position, $P(k + 1)$, is considered.

Case 4 $|\Im_{lim} - \delta\Im| < |\Im(k)|$ When the cutting force magnitude, $|\Im(k)|$, exceeds the target force value, its magnitude should be reduced. The maximum desired value for the cutting force is the target force value. The cutting force is set equal target force:

$$|\Im(k)| \quad = \quad |\Im_{lim}| \tag{4.30}$$

The mechanistic force model is inverted to solve for an actual feed rate, $F_a(k)$. The resulting actual feed rate, $F_a(k)$, is the feed rate that should be achieved to guarantee that the target force is attained. The objective is to achieve the actual feed rate, $F_a(k)$, designated F_{aT}, where T stands for target, at the tool position, $P(k)$, designated P_T:

$$\begin{cases} F_{aT} &= F_a(k) \\ P_T &= \sum_{j=1}^{k-1} F_a(j) \cdot T \end{cases} \tag{4.31}$$

To achieve this objective, the actual feed rate may have to be reduced for some previous lags. This is termed the "regular procedure" and is explained in the next paragraph.

Regular Procedure The initial conditions for the tool motion are the tool position, the actual feed rate, and the acceleration of the feed drive at the previous lag $k-1$. A reference index I is introduced for these initial conditions, $I = k - 1$.

$$\begin{cases} P(I) &= \sum_{j=1}^{I-1} F_a(j) \cdot T \\ F_a(I) & \\ F_S(I) & \end{cases} \tag{4.32}$$

To achieve the feed jump from the actual feed rate of the previous lag to the target feed rate, $\Delta F = (F_{aT} - F_a(i))$, a number N_a of time periods is necessary. Another index, N, is introduced to increment the current lag and account for the added periods, $N = I + N_a$. The objective is to determine a new vector of actual feed rates, designated $[F_a^*]$, and a new vector of the commanded feed rates, designated $[F_S^*]$. These vectors have a size equal to N. These vectors are composed of previous values of the actual feed rate or commanded feed rate up to the index I. The remaining values up to the index N are new values:

$$\begin{cases} [F_a^*] &= [F_a(1)\cdots F_a(I)\ F_a^*(I+1)\cdots F_a^*(N)] \\ [F_S^*] &= [F_S(1)\cdots F_S(I)\ F_S^*(I+1)\cdots F_S^*(N)] \end{cases} \tag{4.33}$$

To search for the number of added time periods, N_a, and, consequently, the number, N, an iterative procedure is used. In this procedure, the time is incremented gradually starting from $t = N \cdot T = (I+1)T$. The feed drive system, in the worst case, needs a duration of τ_{total} to achieve a feed jump. The maximum value of N_a is, therefore, equal to the number of coefficients, N_c, of the Acc/Dec FIR filter.

$$N_a \ \leq \ N_c \ \Leftrightarrow \ (N-1) \ \leq \ N_c \tag{4.34}$$

For any given I and N, the unknowns are the commanded feed rate $F_S^*(i)$, for $i \in [I+1:N]$, the actual feed rate $F_a^*(i)$, for $i \in [I+1:N]$, and the tool position $P(N)$ that the feed drive can achieve. The number of unknowns is equal to $2 \cdot (N-1)+1$.

The equations to solve for these unknowns are given by the relationships between actual and commanded feed rate, the expression of the target feed rate, and the current tool position:

$$\begin{cases} F_a^*(i) &= \sum_{j=1}^{i-I} C_j \cdot F_S^*(i-j+1) + \sum_{j=i-I+1}^{N_c} C_j \cdot F_S(i-j+1)\ for\ i \in [I+1:N] \\ F_a^*(N) &= F_{aT} \\ P(N) &= \sum_{j=1}^{I-1} \frac{F_a(j)\cdot T}{60} + \sum_{j=I}^{N-1} \frac{F_a^*(j)\cdot T}{60} \end{cases} \tag{4.35}$$

The number of equations is equal to $N - I + 2$. The number of equations is less than the number of unknowns. The problem is indeterminant. To solve it, the number of unknowns must be reduced to equal the number of equations. To reduce the number of unknowns the commanded feed rate are set equal to F_{ST} for the added lags i, $i \in [I+1:N]$:

$$F_S^*(i) \;=\; F_{ST} \; for \; i \in [I+1:N] \tag{4.36}$$

The system (4.35) becomes:

$$\begin{cases} F_a^*(i) \;=\; \sum_{j=1}^{i-I} C_j \cdot F_{ST} + \sum_{j=i-I+1}^{N_c} C_j \cdot F_S(i-j+1) \; for \; i \in [I+1:N] \\ F_a^*(N) \;=\; F_{aT} \\ P(N) \;=\; \sum_{j=1}^{I-1} \frac{F_a(j) \cdot T}{60} + \sum_{j=I}^{N-1} \frac{F_a^*(j) \cdot T}{60} \end{cases} \tag{4.37}$$

The system (4.37) can be solved for the commanded feed rate, F_{ST}:

$$F_{ST} \;=\; \frac{F_{aT} - \sum_{j=N-I+1}^{N_c} C_j \cdot F_S(N-j+1)}{\sum_{j=1}^{N-I}} \tag{4.38}$$

The commanded feed rate, F_{ST}, must be within the limits of the feed drive system. The initial objective was to decrease the actual feed rate. The only violation that can occur is that the commanded feed rate, F_{ST}, is less than the minimum value of the feed rate, $F_{ST} < F_{min}$. Two cases are considered:

- The commanded feed rate is less than the minimum allowable feed rate, $F_{ST} < F_{min}$;
- The commanded feed rate is greater than or equal to the minimum allowable feed rate, $F_{ST} \geq F_{min}$.

Case 1 : $F_{ST} < F_{min}$ The feed drive system will overshoot the target feed rate F_{aT}, $F_a^*(N) \geq F_{aT}$. More time periods are needed to achieve the feed jump ΔF, $\Delta F = (F_{aT} - F_a(I))$. The number N is increased and the regular procedure is repeated.

Case 2 : $F_{ST} \geq F_{min}$ The commanded feed rate, F_{ST}, does not violate the lower limit of the feed drive system. The target feed rate, F_{aT}, is achieved, $F_a^*(N) = F_{aT}$. The target position, P_T, must be checked. The values of the actual feed rate are evaluated, and the tool position, $P(N)$, is estimated:

$$\begin{aligned} F_a^*(i) &= \sum_{j=1}^{i-I} C_j \cdot F_{ST} + \sum_{j=i-I+1}^{N_c} C_j \cdot F_S(i-j+1) \\ & \quad for\ i \in [I+1 : N-1] \end{aligned} \tag{4.39}$$

$$P(N) = \sum_{j=1}^{i-I} \frac{F_a(j) \cdot T}{60} + \sum_{j=I}^{N-1} \frac{F_a^*(j) \cdot T}{60}] \tag{4.40}$$

Given a certain tolerance interval on the tool location, δP, the tool position, $P(N)$, can be:

- Equal to the target position, $P_T - \delta P \leq P(N) \leq P_T + \delta P$;
- Greater than the target position, $P_T - \delta P < P(N)$;
- Less than the target position, $P(N) < P_T + \delta P$.

Case 2-1 : $P_T - \delta P \leq P(N) \leq P_T + \delta P$; The tool position, $P(N)$, is equal to the target position, P_T. The target position and the target feed rate are satisfied.

The intermediate values of the actual feed rate, $\{F_a^*(I+1)\cdots F_a^*(N-1)\}$, may not satisfy the force constraint for one or more lags. The mechanistic force model is invoked to evaluate the force magnitude for each lag, i, $i \in [I+1 : N]$. If the force constraint is satisfied for all the lags, the time is incremented and the next tool position, $P(K+1)$, is considered. If the cutting force exceeds the target force for a lag i, $i \in [I+1 : N]$, the solution is to reduce the cutting force magnitude for this lag by reducing the actual feed rate for the initial condition, $F_a(I)$. The current lag, k, is set equal to i. The maximum desired value for the cutting force is the target force value. The mechanistic force model is inverted to solve for an actual feed rate, $F_a(k)$. The objective is to achieve the actual feed rate, $F_a(k)$, designated F_{aT}, at the tool position, $P(k)$, designated P_T. The index I is decremented, $I = I - 1$. If the index I is not equal to zero, the regular procedure is repeated. If the index I is equal to zero, an approach procedure is applied. The approach procedure is explained in the next paragraph.

Case 2-2 : $P_T - \delta P < P(N)$; The tool position, $P(N)$, is greater than the target position, P_T. The target position is overshot. The solution is to reduce the actual feed rate for the initial condition, $F_a(I)$. The index, I, is decremented, $I = I - 1$. If the index, I, is not equal to zero, the regular procedure is repeated. If the index, I, is equal to zero, an approach procedure is applied.

Case 2-3 : $P(N) < P_T + \delta P$; The tool position, $P(N)$, is less than the target position, P_T. The target position is undershot. The target feed rate is achieved in a number of added periods equal to $N_a = N - I$. By examining the number N_a, there are two conditions that can occur:

- If the number of the added periods, N_a, is less than the number of coefficient of the filter, the number of added periods is increased, $N = N + 1$. The regular procedure is repeated.
- If the number, N_a, is equal to the number of the coefficient of the Acc/Dec FIR, $N_a = N_c$, the acceleration at the target position is zero because the feed drive can achieve any feed jump within a time interval equal to τ_{total}. The target position is achieved by adding a number of periods, n, with a constant feed rate. The constant feed rate is equal to $F_a^*(N)$. The number, n, is estimated by:

$$n \;=\; round\left(60 \cdot \frac{|P(N) - P_T|}{F_a^*(N)}\right) \tag{4.41}$$

The vectors of the actual feed rate and commanded feed rate are extended. The intermediate values of the actual feed rate, $\{F_a^*(I+1) \cdots F_a^*(N-1)\ F_a^*(N)\ F_a^*(N) \cdots F_a^*(N+n)\}$, may not satisfy the force constraint for one or more lags. The mechanistic force model is invoked to evaluate the force magnitude for each lag, i, $i \in [I+1 : N+n]$. If the force constraint is satisfied for all the lags, the time is incremented and the next tool position, $P(K+1)$, is considered. If the force constraint is not satisfied for one lag, the solution is to reduce the cutting force magnitude for this lag by reducing the actual feed rate for the initial condition, $F_a(I)$. The current lag, k, is set equal to i. The maximum desired value for the cutting force is the target force value. The mechanistic force model is inverted to solve for an actual feed rate, $F_a(k)$. The objective is to achieve the actual feed rate, $F_a(k)$, designated

F_{aT}, at the tool position, $P(k)$, designated P_T. The index, I, is decremented, $I = I - 1$. If the index, I, is equal to zero, the regular procedure is repeated. If the index, I, is equal to zero, an approach procedure is applied.

The flow chart for the regular procedure is depicted in figure 4.4.

Approach procedure The approach procedure is applied if the feed drive system cannot satisfy all the constraints with a given feed rate ($F_a(1)$,$F_S(1)$). The constraints that may have been violated include the target position is overshot, the force constraint is not satisfied, or the commanded feed rate does not respect the limits of the feed drive system. The necessary conditions are the index, I, for the last iteration is equal to one, and the value of N is less than the number of FIR filter coefficients, N_c. The actual feed rate, $F_a(1)$, is reduced so that the tool reaches the target position, P_T, with the target feed rate, F_{aT}, within a minimum traveling time. The objective is to determine a vector of actual feed rate, designated $[F_a^*]$, and a vector of the commanded feed rate, designated $[F_S^*]$. The actual feed rate must be reduced. Therefore, the feed drive needs more time than the number of periods given by N to achieve the target feed rate at the target position. As a first guess, the size of these vectors is set equal to N. These vectors should guarantee that:

- The actual feed rate respects the Acc/Dec profile of the feed drive. The actual feed rate is determined from the commanded feed rate using the Acc/Dec FIR filter:

$$F_a^*(i) = \sum_{j=1}^{N_c} C_j \cdot F_S^*(i - j + 1) \ for \ i \in [1 : N] \tag{4.42}$$

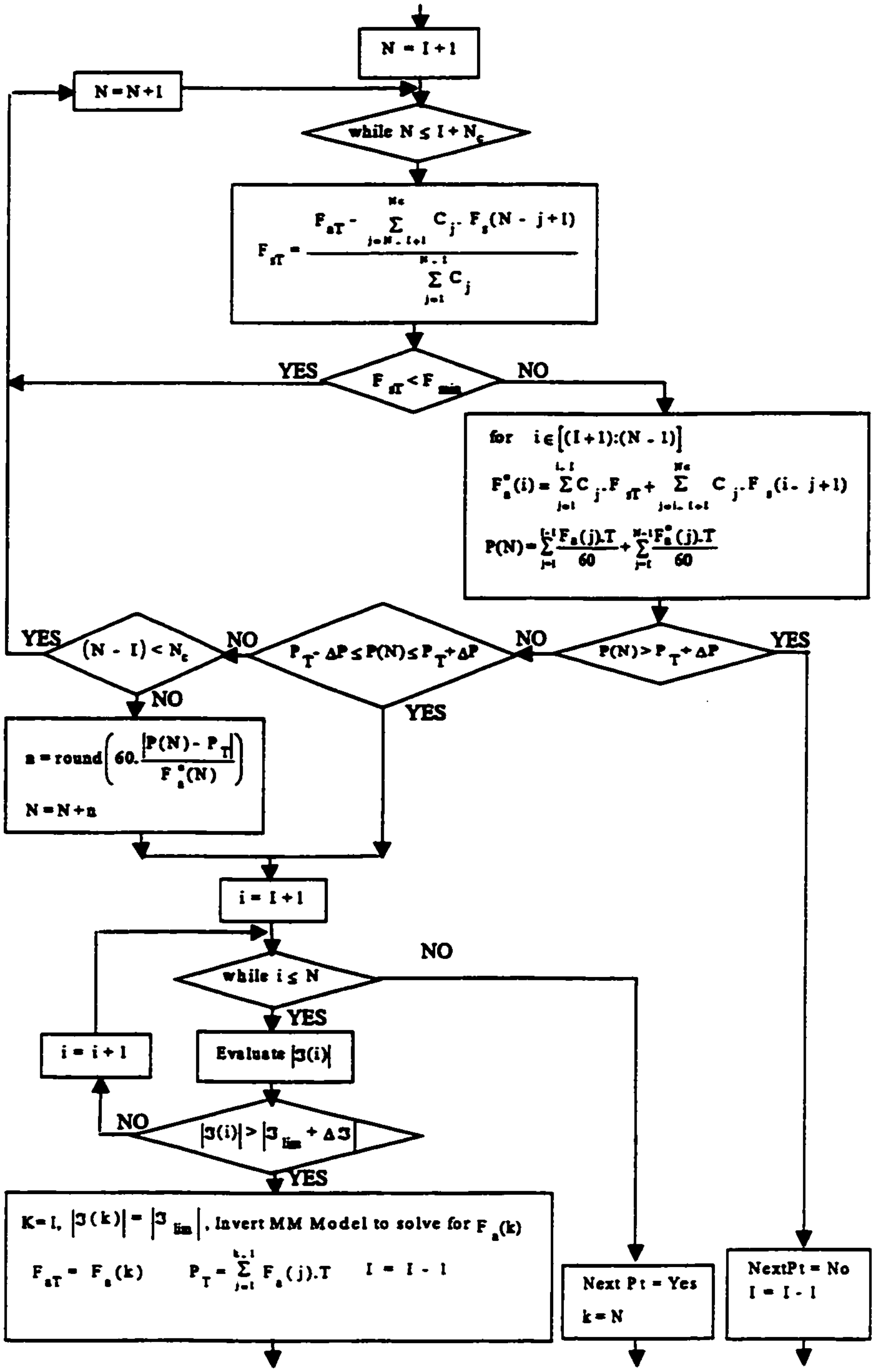


Figure 4.4: Flow chart of the "regular procedure"

- The target position is reached:

$$P_T = \sum_{j=1}^{N-1} F_a^*(j) \cdot \frac{T}{60} \tag{4.43}$$

- The target feed rate is achieved:

$$F_a^*(N) = F_{aT} \tag{4.44}$$

- The number of time periods, N, is minimized to guarantee the shortest traveling time.

For a given number of time periods, N, the unknowns are the commanded feed rate, $F_S^*(i)$, for $i \in [1:N]$, and the actual feed rate, $F_a^*(i)$, for $i \in [1:N]$. The number of unknowns is equal to $2N$. The equations to solve for these unknowns are (4.42), (4.43), and (4.44). The number of equations is equal to $N+2$. N is greater than 2. Therefore, the number of equations is less than the number of unknowns. The problem is indeterminant. To solve it, the number of unknowns must be reduced to equal the number of equations. To reduce the number of unknowns, the commanded feed rate is divided into two sections with constant value. The commanded feed rates are designated by F_{S0} and F_{ST}.

$$\begin{cases} [F_S^*(i)] = [F_S^*(1) \cdots F_S^*(I) \; F_S^*(I+1) \cdots F_S^*(N)] \\ F_S^*(k) = F_{S0} \; for \; k \in [1:I] \\ F_S^*(k) = F_{ST} \; for \; k \in [I+1:N] \end{cases} \tag{4.45}$$

The index I introduced in equation (4.45) is also unknown. It is determined in an iterative way. For any given I and N, the values of the commanded feed rate, F_{S0} and F_{ST}, are determined using equations (4.42), (4.43), (4.44), and (4.45). The expressions for F_{S0} and F_{ST} are different, depending on the value of $N - I$. When $N - I$ is less than the number of the FIR filter coefficients N_c, the target feed rate, F_{aT}, will not reach the commanded feed rate, F_{ST}. The expressions for F_{S0} and F_{ST} are:

$$F_{S0} = \left[\frac{60P_T}{T} \cdot S_1 - S_3 \cdot F_{aT}\right] \cdot \frac{1}{S_1 \cdot S_4 - S_2 \cdot S_3} \tag{4.46}$$

$$F_{ST} = \frac{F_{aT} - \sum_{j=N-I+1}^{min(N_c,N)} C_j \cdot F_{S0}}{\sum_{j=1}^{N-I} C_j} \tag{4.47}$$

where: $S_1 = \left(\sum_{j=1}^{N-I} C_j\right)$, $S_2 = \left(\sum_{j=N-I+1}^{min(N_c,N)} C_j\right)$, $S_3 = \left(\sum_{i=I+1}^{N-1} \sum_{j=1}^{i-I} C_j\right)$, and $S_4 = \left(\sum_{i=I+1}^{N-1} \sum_{j=i-I+1}^{min(N_c,i)} C_j + \sum_{i=1}^{I} \sum_{j=1}^{min(N_c,i)} C_j\right)$. If $N - I$ is greater than or equal to the number of the FIR filter coefficients, $N - 1 \geq N_c$, the target feed rate, F_{aT}, will reach the commanded feed rate, F_{ST}. The expressions for F_{S0} and F_{ST} are:

$$F_{S0} = \left[\frac{60P_T}{T} - S_5 \cdot F_{aT}\right] \cdot \frac{1}{S_6} \tag{4.48}$$

$$F_{ST} = F_{aT} \tag{4.49}$$

where: $S_5 = \left(\sum_{i=I+1}^{I+N_c-1} \sum_{j=1}^{i-I} C_j + \sum_{i=I+N_c}^{N-I} 1\right)$, and $S_6 = \left(\sum_{i=1}^{I} \sum_{j=1}^{min(N_c,i)} C_j + \sum_{i=I+1}^{I+N_c-1} \sum_{j=i-I+1}^{min(N_c,i)} C_j\right)$. In both cases, F_{S0} is greater than F_{ST}. To save production time, the solution where the most traveling time is generated by F_{S0} is preferred. The Index, I, is initialized to $N-1$ to guarantee the minimum time of traveling. The commanded feed rate, F_{S0} and F_{ST} , should be within the limits of the feed drive system:

$$\begin{cases} F_{min} \leq F_{S0} \leq F_{max} \\ F_{min} \leq F_{ST} \leq F_{max} \end{cases} \tag{4.50}$$

If the commanded feed rates, F_{S0} or F_{ST} , violates one of the constraints given by equation (4.50), the actual feed rate must be reduced. To guarantee minimum traveling time, the number of lags where the commanded feed rate, F_{S0}, is effective must be reduced. The index, I, is decreased and the iterations to find a valid I continue. If the index, I, equals one, the index, N, is increased and the ,approach procedure is repeated.

If the commanded feed rates, F_{S0} and F_{ST}, do not violate these constraints, the mechanistic model is invoked to evaluate the force magnitude for each lag, i, $i \in [1:N]$. If the force constraint is respected for all lags, then the time is incremented and the next tool position, $P(k+1)$, is considered. If the force constraint is violated for one lag, the index, I, should be decreased and the iterations to find a valid I should continue. If the Index, I, equals one, the index, N, is increased and the approach procedure is repeated.

The approach procedure results in the appropriate value of the number of lags, N, the appropriate value of the Index, I, the commanded and the actual feed rate, the tool position and the cutting force for lags i, $i \in [1:N]$. The flow chart for this procedure is shown in figure 4.5.

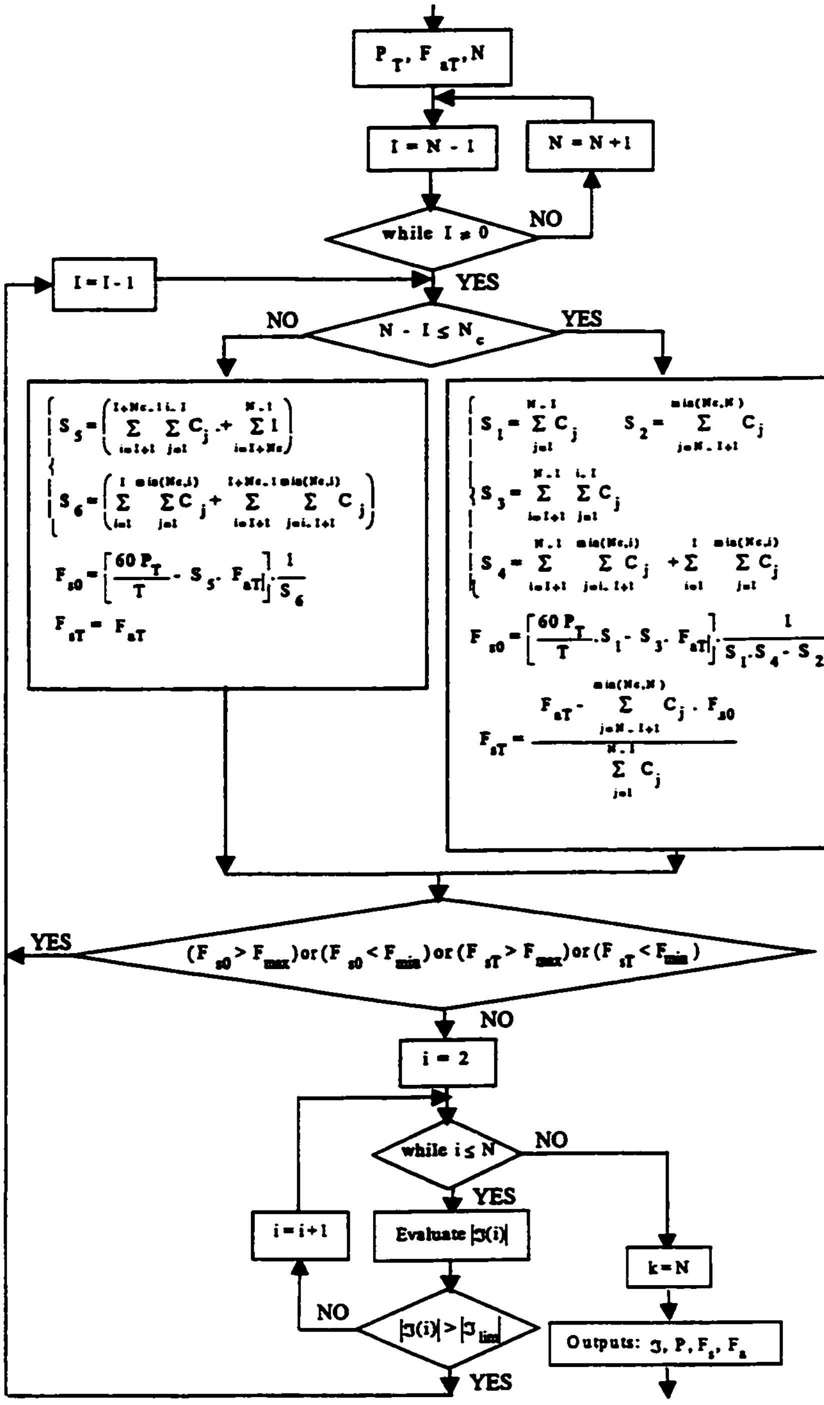


Figure 4.5: Flow chart of the “approach procedure”

Generation of the feed drive deceleration at the end of the tool path

When the current tool position exceeds the tool path length, the commanded feed rate should be reduced for some previous lags to allow the deceleration of the feed drive to reach zero velocity and zero acceleration at the end of the tool path. At the end of the path, the tool target position is designated by P_T. The actual feed rate and the commanded feed rate are designated as F_{aT} and F_{ST}, respectively, and are both null. To decelerate from any given feed rate to zero velocity requires an interval of time less than or equal to the Acc/Dec time constant, τ_{total}. The objective is to determine the commanded feed rate vector, $[F_S^*]$, and the actual feed rate vector, $[F_a^*]$, so that the target position, P_T, is reached with the target feed rate, F_{aT}, within the minimum traveling time. These vectors have size N. These vectors should guarantee that:

- The actual feed rate respects the Acc/Dec profile of the feed drive. The actual feed rate is determined from the commanded feed rate using the Acc/Dec FIR filter:

$$F_a^*(i) = \sum_{j=1}^{N_c} C_j \cdot F_S^*(i-j+1) \; for \; i \in [I+1:N] \tag{4.51}$$

- The target position is reached:

$$P_T = \sum_{j=1}^{N-1} F_a^*(j) \cdot \frac{T}{60} \tag{4.52}$$

- The target feed rate is equal to zero in order to stop at the end of the path. The feed drive system requires an interval of time equal to the Acc/Dec time constant, τ_{total}, in order to decelerate from any given feed rate to zero velocity. The commanded feed rate is, therefore, set equal to zero for the last N_c lags:

$$\begin{cases} F_a^*(N) = F_{aT} = 0 \\ F_S^*(i) = F_{ST} = 0 \;\; for\ i \in [N - N_c + 1 : N] \end{cases} \tag{4.53}$$

- The number of time periods, N, is minimum in order to guarantee the shortest traveling time. For a given number of time periods, N, and a given Index, I, the unknowns are the commanded feed rate, $F_S^*(i)$, for $i \in [I+1:N]$, and the actual feed rate, $F_a^*(i)$, for $i \in [I+1:N]$. The number of unknowns is equal to $2(N-I)$. The equations to solve for these unknowns are (4.51), (4.52), and (4.53). The number of equations is equal to $N - I + 2 + N_c$. The number of equations is less than the number of unknowns. The problem is indeterminant. To solve it, the number of unknowns must be reduced to equal the number of equations. To reduce the number of unknowns, the commanded feed rate is assumed to be constant and equal to F_{S0} for the lags i, $i \in [I+1:r]$ where $r = N - N_c$:

$$\begin{cases} [F_S^*] = [F_S^*(1) \cdots F_S^*(I)\ F_S^*(I+1) \cdots F_S^*(r)\ F_S^*(r+1) \cdots F_S^*(N)] \\ F_S^*(k) = F_S(k)\ for\ k \in [1:I] \\ F_S^*(k) = F_{S0}\ \ for\ k \in [I+1:r] \\ F_S^*(k) = F_{ST} = 0\ for\ k \in [r+1:N] \\ r = N - N_c \end{cases} \tag{4.54}$$

The index I is also an unknown. The index I is determined in an iterative way.

The value of the commanded feed rate, F_{S0}, is determined using the equations (4.51), (4.52), (4.53), and (4.54). F_{S0} is expressed as:

$$\begin{aligned} F_{S0} &= \frac{\left[\frac{60P_T}{T} - \sum_{i=1}^{I} F_a^*(i) - S_8\right]}{S_7} \qquad (4.55) \\ S_7 &= \left(\sum_{i=I+1}^{r} \sum_{j=1}^{i-I} C_j + \sum_{i=r+1}^{N-1} \sum_{j=i-r+1}^{min(N_c, i-I)} C_j \right) \\ S_8 &= \left(\sum_{i=I+1}^{r} \sum_{j=i-I+1}^{min(N_c, i)} C_j \cdot F_S^*(i-j+1) \right. \\ &\quad \left. + \sum_{i=r+1}^{N-1} \sum_{j=i-I+1}^{min(N_c, i)} C_j \cdot F_S^*(i-j+1) \right) \end{aligned}$$

Once the commanded feed rate is evaluated, three cases have to be considered:

- The commanded feed rate is less than the minimum allowable feed rate, $F_{ST} < F_{min}$. The commanded feed rate, F_{ST}, violates the lower limit of the feed drive system. The index, I, is decreased. Then the deceleration procedure is repeated.
- The commanded feed rate is greater than the maximum allowable feed rate, $F_{ST} > F_{max}$. The commanded feed rate, F_{ST}, violates the upper limit of the feed drive system. The number of time periods, N, is increased.
- The commanded feed rate is within the limits of the feed drive, $F_{min} \leq F_{ST} \leq F_{max}$. The feed rate set, F_{ST}, does not violate the limits of the feed drive system. The commanded feed rate, the actual feed rate, and the cutting force is evaluated for the lags i , $i \in [I+1 : N]$. The generation of the tool motion is completed.

The flow chart for the generation of the feed drive deceleration at the end of the tool path is shown in figure 4.6.

4.3.3 Generation of the G-Code

Each motion block k in the G-Code file is composed of a commanded feed rate $F_S(k)$, and a move distance $\Delta X(k)$. When more than one move is specified in succession, with no pause in between, the first move will be blended into the second move according to the Acc/Dec time parameters: τ_{acc} and $\tau_{dec} = \tau_{total} - \tau_{acc}$ [13]. These parameters are specified in appendix D. For blended motion, the CNC controller will determine the time for the move, designated $T_M(k)$, based on the commanded feed rate and the move distance:

$$T_M(k) = \frac{\Delta X(k) \cdot 60}{F_S(k)} \tag{4.56}$$

The commanded feed rate $F_S(k)$, for $k \in [1 : N]$, used to generate the tool motion is determined as a function of time $t = k \cdot T$. The time period T corresponds to the time for the move, $T_M = T$. The move distance is determined from:

$$\Delta X(k) = \frac{F_S(k) \cdot T}{60} \tag{4.57}$$

This expression is effective while the commanded feed rate is not null. During the deceleration phase at the end of the path, the commanded feed rate has been set to zero for a number of lags equal to the number of the coefficients of the Acc/Dec FIR filter. The characteristics of the last move are:

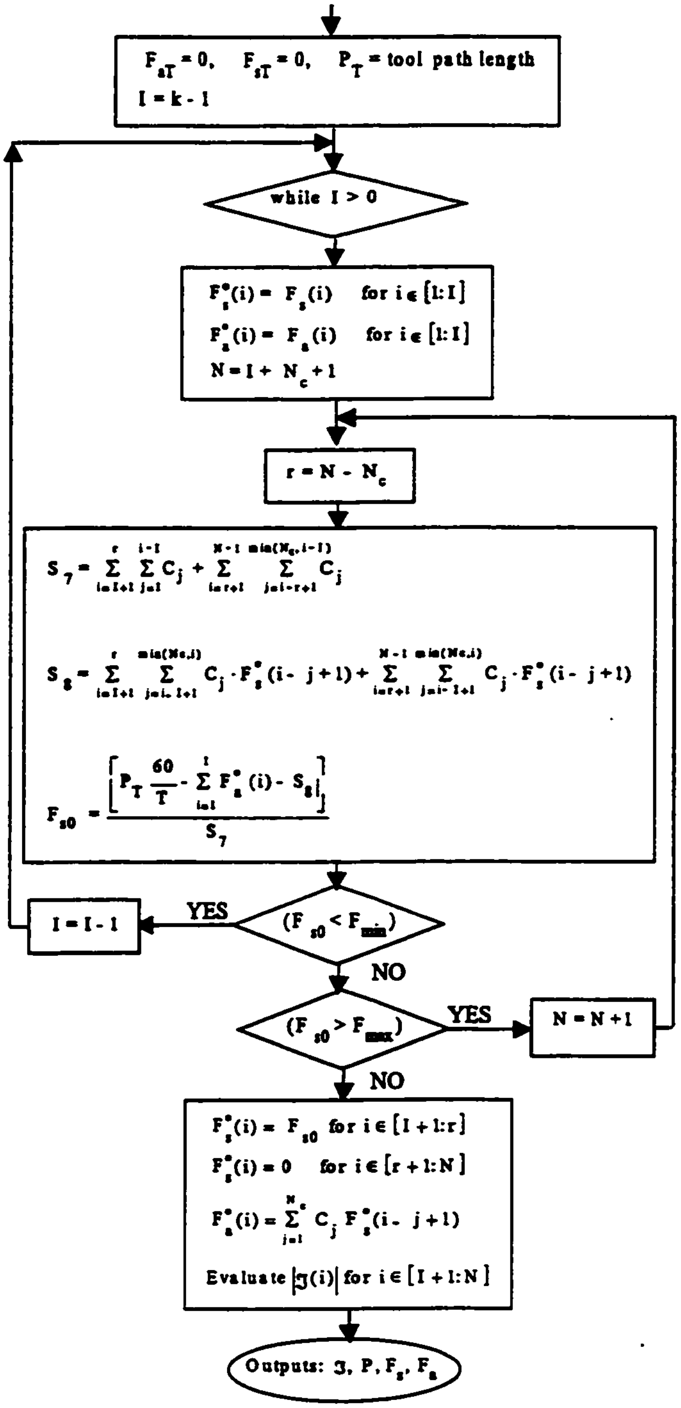


Figure 4.6: Flow chart for the "generation of the feed drive deceleration at the end of the tool path"

- The move distance is equal to the distance that the tool travels in the deceleration phase.

$$\Delta X(N - N_c + 1) = \sum_{i=N-N_c}^{N} \frac{F_a(i) \cdot T}{60} \tag{4.58}$$

- The commanded feed rate is equal to the commanded feed rate value immediately before the deceleration phase:

$$F_S(N - N_c + 1) = F_S(N - N_c) \tag{4.59}$$

The resulting number of the motion blocks is $N - N_c + 1$. The number of motion blocks is dependent on the sampling period and the tool path length. Combining successive motion blocks with an identical commanded feed rate could reduce the number of blocks and, consequently, reduce the size of the G-Code file.

4.4 Summary

A feed scheduling methodology for multi-axis machining was presented in this chapter. Feed rate scheduling is performed at user specified intervals along the given tool path (control points). For each control point, the cutting geometry and cutting forces are extracted using the modeling methodologies presented in chapter 3. The maximum allowable feed rate is scheduled by applying a process constraint, either maximum chip load or maximum force.

A methodology for integrating the machine tool feed drive performance capabilities with feed scheduling was also presented in this chapter. Two strategies were presented.

The first strategy considered the feed drive Acc/Dec time constant for feed scheduling. The second strategy optimizes the feed drive capabilities for feed scheduling. This strategy integrates the feed drive performance capabilities with the generation of the tool path in order to achieve the desired feed rate at the appropriate tool position.

Chapter 5

Force Model Calibration and Verification Results

5.1 Introduction

This chapter provides an overview of the simulations and experimental verification for the modeling methodologies presented in chapter 3. The force model calibration results are presented in section 5.2. The force model verification results are given in section 5.3. An overview of some of the simulation capabilities is discussed in section 5.4. The dynamic process simulation methodology is demonstrated in section 5.5. Applications of the Internet based machining system are discussed in section 5.6. Finally, a summary is offered in section 5.7.

5.2 Calibration Results

The calibration procedure detailed in section 3.4 was used to determine the cutting force coefficients (K_1, K_2, K_3, K_4, K_5) for a set of ball end milling experiments. The tool used was a 330M Garr 3/4" diameter, solid carbide ball end mill. The cutting edge for this tool was a custom grind. Therefore, the cutting edge profile had to be determined from measurements taken using a CMM. The workpiece material was a commercial Titanium (Ti6Al 4V). The experiments were divided into a series of cutting tests that varied the effective cutting speed, V_j, and helix angle, β_j , for a constant feed rate. This variation in V_j and β_j was achieved by cutting over different radial and axial immersion of the ball part of the tool. The ball section was divided into five regions, each having equal included angles.

A range of feed rates were also selected for calibrating the tool (0.003",0.004", 0.005",0.006" feed per tooth). At each feed rate, a cutting test was performed for each of the five regions of the ball described above, the total number of calibration tests being 20.

For each test, the average calibration coefficient, K_i, was determined using calibration procedure step 3. The coefficients having constant feed per tooth were then combined, and equation 3.28 was fit to these combined coefficients using calibration procedure step 4. Finally, the parameter values a_1, a_2, and a_3 (for each K_i) were fit over the range of feed rates using calibration procedure step 5. Figures 5.1, 5.2, 5.3, 5.4, and 5.5 show the results of the calibration procedure for cutting coefficients K_1, K_2, K_3, K_4, K_5 respectively. The units for K_1, K_2, and K_3 are N/mm^2, and for K_4 and K_5 are N/mm. In each figure; the first graph (a) shows trends for a feed per tooth of 0.003". The average coefficient (determined by calibration step 3) are shown

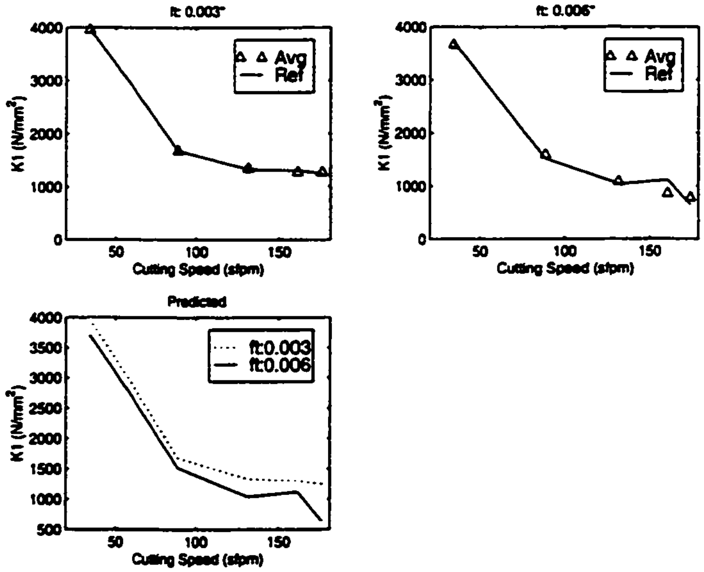


Figure 5.1: Calibration Coefficient K_1

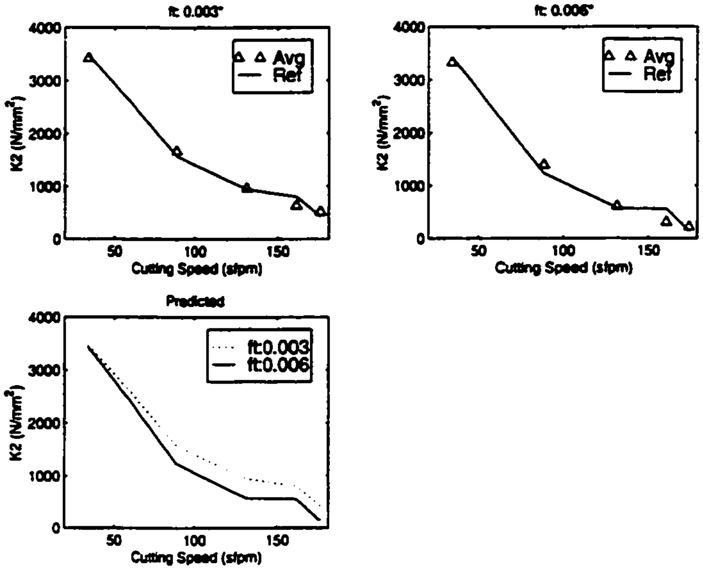


Figure 5.2: Calibration Coefficient K_2

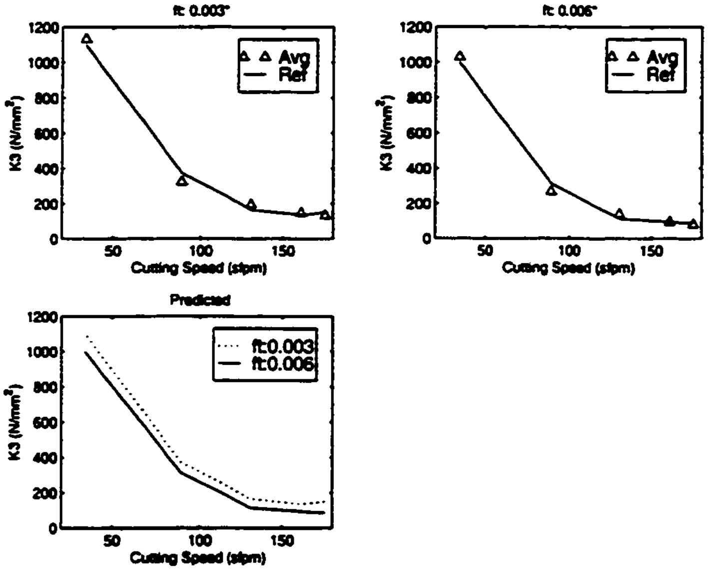


Figure 5.3: Calibration Coefficient K_3

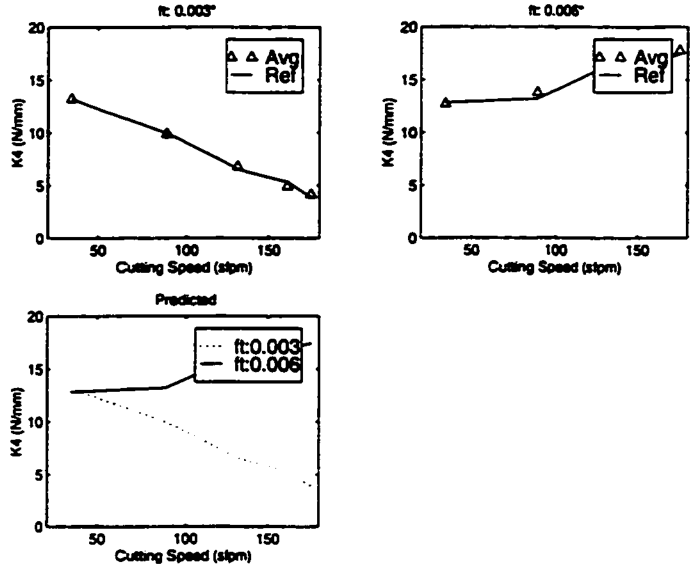


Figure 5.4: Calibration Coefficient K_4

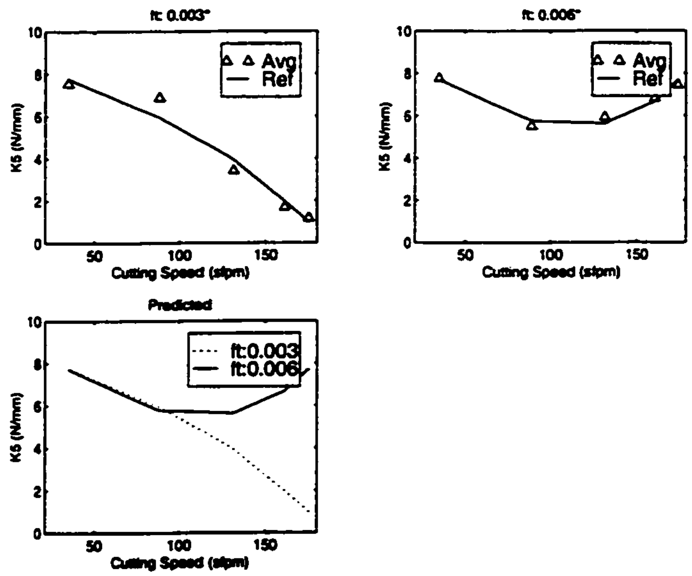


Figure 5.5: Calibration Coefficient K_5

as points with (△). The coefficient determined using equation 3.28 in calibration step 4 are shown as solid lines. The second graph (b) shows trends for a feed per tooth of 0.006". The average coefficient (determined by calibration step 3) are shown as points with (△). The coefficient determined using equation 3.28 in calibration step 4 are shown as solid lines. The third graph (c) shows a plot of the predicted coefficient using equation 3.28 for both feed rates, 0.003" and 0.006".

In Graph a, and b of each figure the average coefficients are determined for each specific cutting test (ie. constant cutting velocity and helix angle). The cutting coefficients determined by equation 3.28 incorporate the effect of varying cutting velocity and helix angle. The fit obtained by using equation 3.28 is quite good for all K_i. The third graph (c) in each figure demonstrates the effect of varying the feed rate. For K_1, K_2, and K_3, an increase in feed rate causes a decrease in the cutting force coefficient. For K_4 and K_5 an increase in feed rate causes an increase in cutting force coefficient.

5.3 Force Model Verification Results

To verify the force predictions for the modeling methodology, a series of cutting tests were performed. The test conditions were selected to vary the cutting conditions (cutting speed, feed rate, tool immersion geometry). The results presented here have the following cutting conditions; 1/4 radial and axial immersion with a feed per tooth of 0.0045" and a cutting speed of 215 sfpm. The cutting force coefficients determine in the previous section are used for the force prediction. Figure 5.6 shows a cutting force profile for a single tooth. The experimentally measured force is shown as dots, the predicted force using an average coefficient as a thin line, and the predicted force using coefficients determined using equation 3.28 as a thick line. The force profile predicted using an average coefficient fits the experimental data better than the force profile predicted using equation 3.28. This better fit is a bit misleading because the average coefficients are only valid for this specific test. The coefficients determined by using equation 3.28 are valid over a range of cutting conditions. Therefore, calibration coefficients determined using equation 3.28 have a wider, more practical range of application. The multi-axis force model holds cutting force prediction within 5%.

5.4 Simulation Capabilities

The multi-axis force model can also be used to predict dynamic cutting forces and static/dynamic tool deflection. A set of simulations were performed to demonstrate

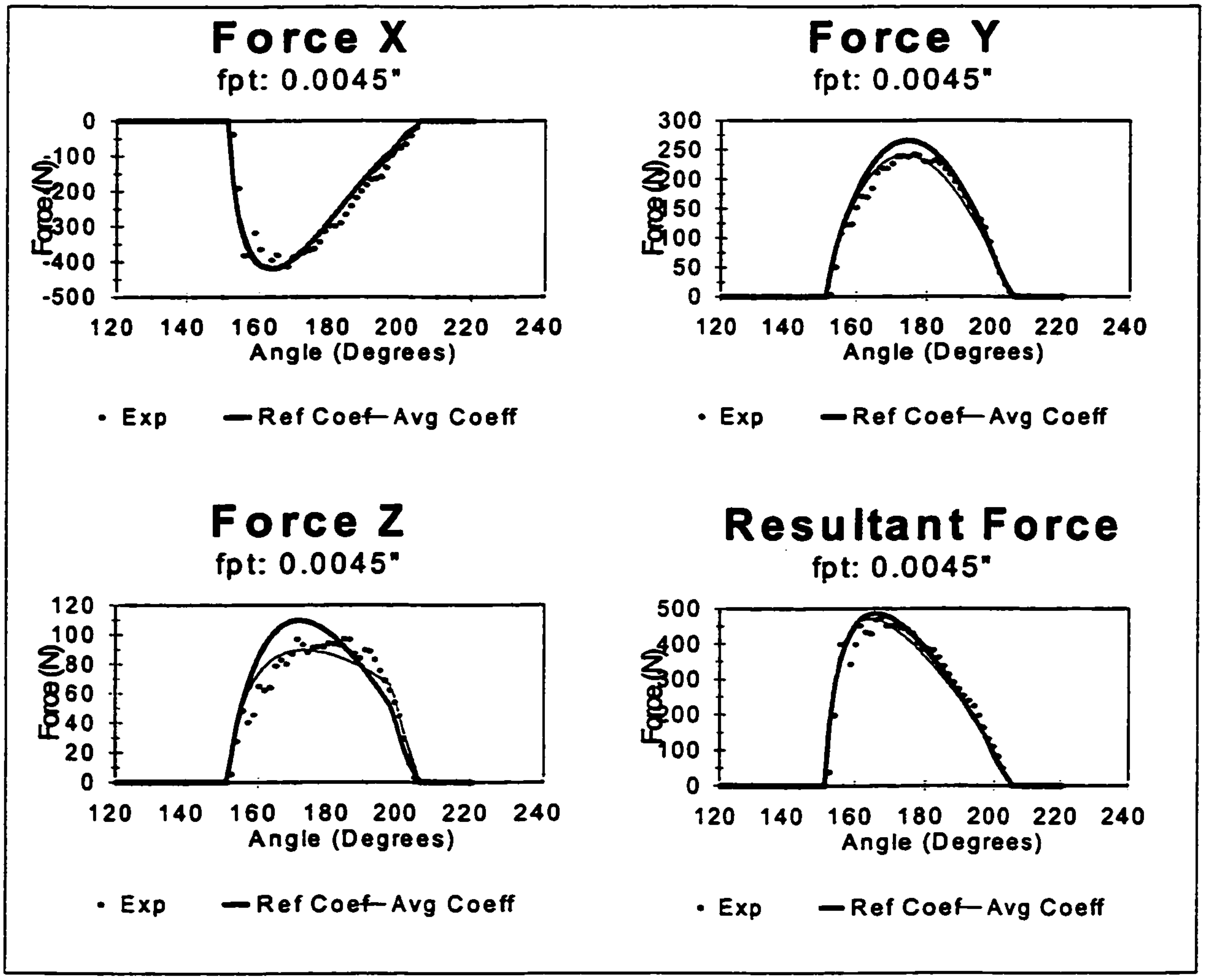


Figure 5.6: Force Model Verification Test

Test	Rad. Imm	Constant
1	1/8	fpt: 0.004" Ax. Imm: 0.75" Dia: 1.0" Tool Len: 2.5"
2	1/4	
3	3/8	

Table 5.1: Static Deflection Test Conditions

the effect of tool radial immersion on predicted process values. The first simulation is used to demonstrate the effect of variable radial immersion on tool static deflection. Tool deflections in both the normal and tangent directions relative to the tool path are predicted. The test conditions simulated variable radial immersion for flat end milling. Test conditions are summarized in table 5.1. The cutting force coefficients determine in the previous section are used. Figure 5.7 shows the predicted tool deflections for the different radial immersions. Figure 5.7a shows tool deflection normal to the tool path. Figure 5.7b shows tool deflection tangent to the tool path. As the radial immersion increases, both the normal and tangential tool deflection increases. The second simulation is used to demonstrate how the modeling methodology can be used for predicting dynamic cutting forces and tool deflection. A flat end mill is used. The test conditions are selected to vary the radial immersion. The cutting force coefficients determine in the previous section are used. Figure 5.8a and 5.8b shows predicted dynamic forces for 1/8 and 1/4 radial immersion respectively. Figure 5.9 shows the corresponding predicted dynamic deflections. Figure 5.9a and 5.9b are the dynamic deflections in the x and y directions for 1/8 radial immersion. Figure 5.9c and 5.9d are the dynamic deflections in the x and y directions for 1/4 radial immersion.

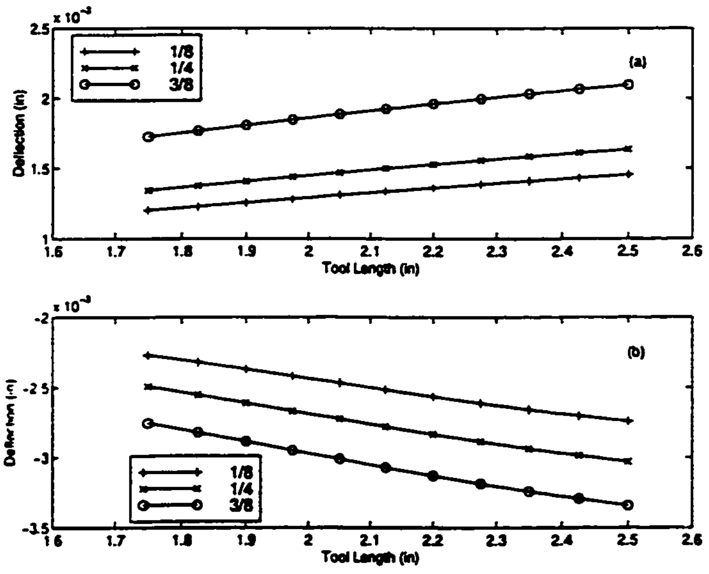


Figure 5.7: Static Tool Deflections for various radial immersions a) Normal b) Tangential

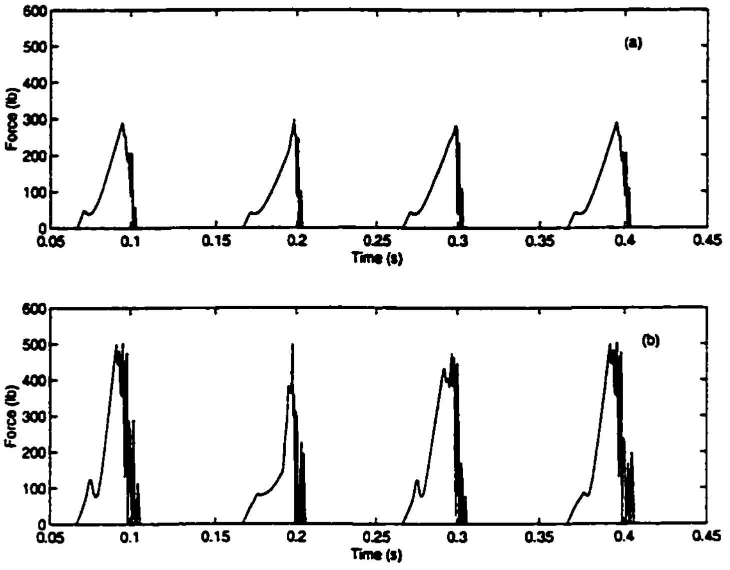


Figure 5.8: Predicted Dynamic Forces a)1/8 rad imm b)1/4 rad imm

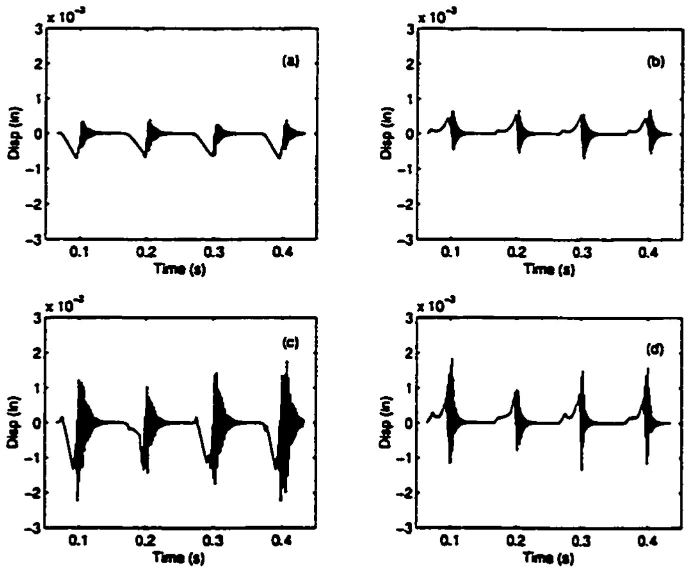


Figure 5.9: Predicted Dynamic Deflection (x,y directions)

5.5 Dynamic Modeling Methodology

This section demonstrates the enhanced approach for dynamic mechanistic modeling methodology for multi-axis machining. The outline for this section is as follows. Section 5.5.1 compares the simulation capabilities for simple linear cuts for the static and dynamic implementation based on control points and the continuous implementation. Section 5.5.2 demonstrates the continuous dynamic process simulation methodology for a complex sculptured surface part.

5.5.1 Dynamic Modeling Methodology Comparison

The modeling methodologies described in sections 3.6, 3.7, and 3.8 are compared here for linear cuts. All simulations are performed with a four flute 330M Garr 3/4" diameter, solid carbide ball end mill. The workpiece material was Titanium (Ti6Al4V).

Cutting experiments are used to determine the force calibration constants (table 5.2). The modal parameters; equivalent mass, damping and stiffness, are determined experimentally using impact tests (table 5.3).

Rake face			Clearance Face	
K_1 *(lb/inch²)*	K_2 *(lb/inch²)*	K_3 *(lb/inch²)*	K_4 *(lb/inch)*	K_5 *(lb/inch)*
152176	199736	3610	39.06	26.18

Table 5.2: Force Calibration coefficients

The first set of simulations are performed for a linear cut with constant axial depth and radial width of cut. The axial depth of cut is 0.1546" and the radial width of cut is 0.1546". The length of cut is 0.08" (4 revolutions of tool). The feed rate is 18.0 ipm and the spindle speed is 900 rpm. Figures 5.10, 5.11, and 5.12 show the simulation results. Figures 5.10 and 5.11 show the force results for the last revolution of the tool. Figure 5.12 show the deflection results for 4 revolutions of the tool. The X force components are shown in figure 5.10. Figure 5.10a is the X force component for the static force modeling methodology. Figure 5.10b is the X force component for the dynamic modeling methodology using control points. Figure 5.10c is the X force component for the continuous dynamic modeling methodology. The Y force is shown in figure 5.11. Figure 5.11a is the Y force for the static force modeling

	X	Y
mass (lb)	0.00725	0.006975
stiffness (lb/in)	154203	114203
damping (lb/s)	1.639943	2.454205

Table 5.3: Modal Parameters

methodology. Figure 5.11b is the Y force for the dynamic modeling methodology using control points. Figure 5.11c is the Y force for the continuous dynamic modeling methodology. The X and Y deflection components are shown in figure 5.12. Figure 5.12a and 5.12b are the X and Y deflection components for the dynamic modeling methodology using control points. Figure 5.12c and 5.12d are the X and Y deflection components for the continuous dynamic modeling methodology.

Experimental forces are overlayed with the predicted forces for the different modeling methodologies in both figure 5.10 and figure 5.11. In figures 5.10a and 5.11a the static modeling methodology predicts the general force profile but does not show any dynamic effects. In figures 5.10b and 5.11b the dynamic modeling methodology using control points predicts both the force profile and dynamic effects. In figures 5.10c and 5.11c the continuous dynamic modeling methodology also predicts both the force profile and dynamic effects. It is evident from these figures that both the control point methodology and the continuous methodology are capable of modeling machining dynamics for constant immersion cuts. It is also evident from these figures that there is a difference in the dynamic behavior of the tool between the two methodologies. The continuous methodology captures more of the process dynamics. This difference is a direct result of how the tool incut segments are determined. The control point methodology calculates the tool incut segments based on a static tool/workpiece model. The control point methodology does not consider the effect of instantaneous tool deflection on the tool incut segments. The continuous methodology calculates the tool incut segments based on the instantaneous location of the cutting edge. The instantaneous location of the cutting edge is determined by considering both the tool centre position and the deflection of the tool.

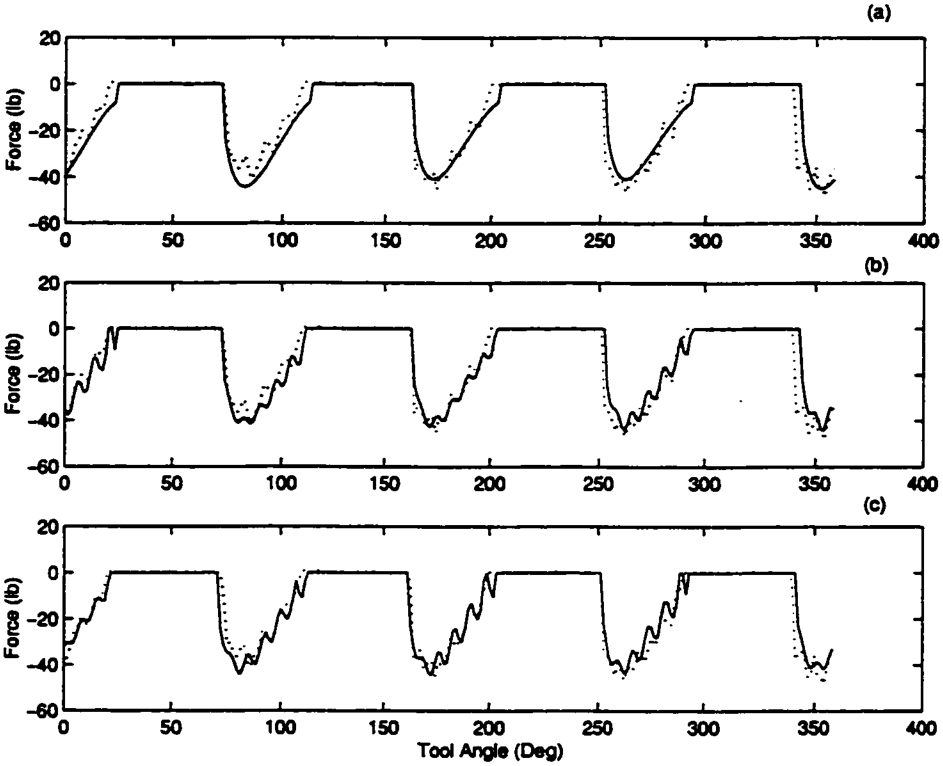


Figure 5.10: X Force component

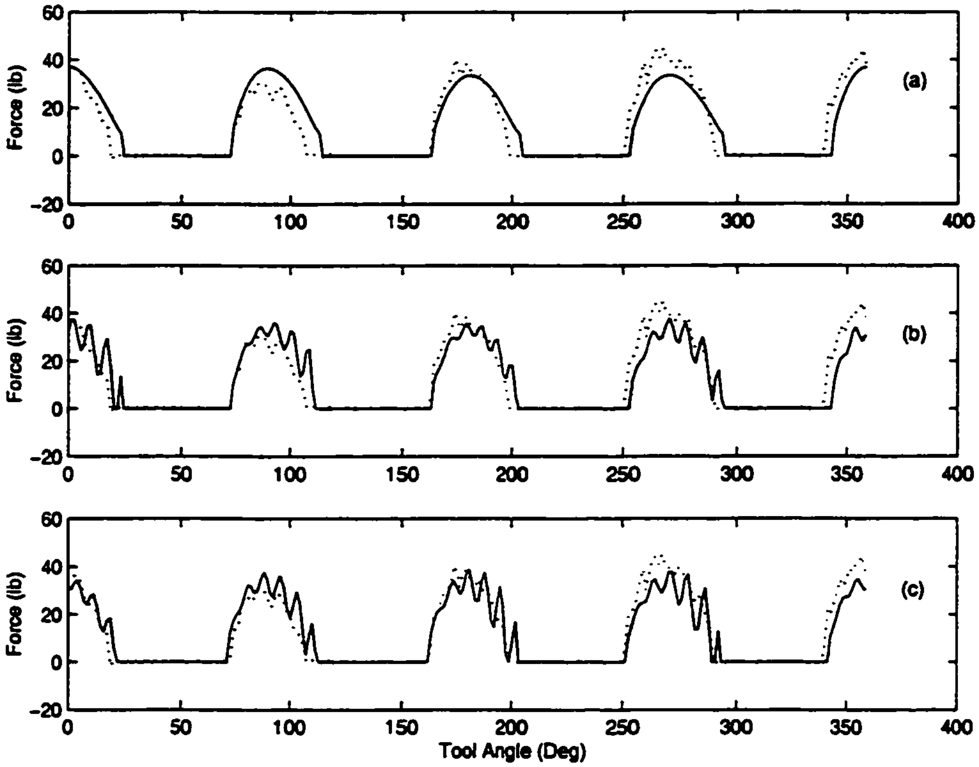


Figure 5.11: Y Force component

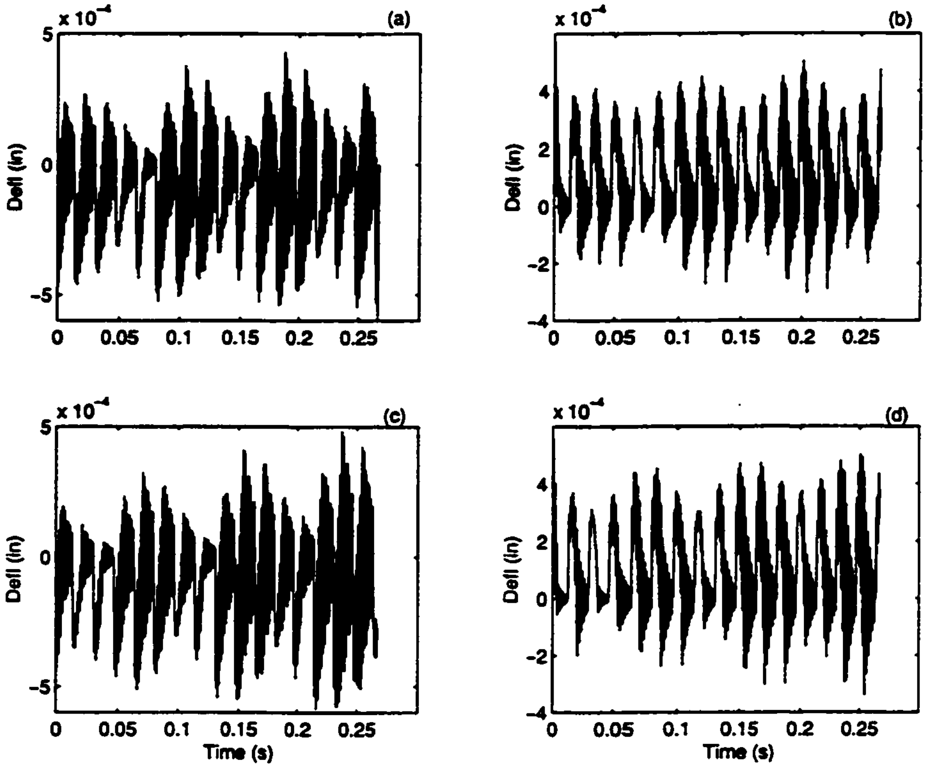


Figure 5.12: X/Y Tool Deflection

The second set of simulations are performed for a linear cut with constant axial depth of cut and variable radial width of cut. The axial depth of cut is 0.125" and the radial width of cut varies between 0.25" and 0.314". The length of cut is 0.128" (16 revolutions of tool). The feed rate is 9.6 ipm and the spindle speed is 1200 rpm. Figure 5.13 shows the simulated resultant cutting force over the entire tool path. Figure 5.14 shows the resultant force in the range between 10 and 14 tool revolutions.

Figure 5.13a and 5.14a show the resultant force for the dynamic modeling methodology using control points. The control point spacing along the tool path is set equal to 0.016". This distance corresponds to the distance travelled during two revolutions of the tool. At each control point the dynamic simulation is performed for two revolutions of the tool. The dynamic simulation for each control point are combined to represent the dynamic simulation over the entire tool path. The dynamic simulation over the entire tool path is considered piece-wise continuous.

These figures shows that there is a step-wise increase in the static and dynamic component of the cutting force for each control point. The step increase in cutting force is a result of the incut segment calculation and the tool deflection history. At each control point the incut segments are determined assuming a static tool/workpiece model. The incut segments are also calculated assuming constant tool/workpiece immersion over the tool path interval between control points. When the immersion of the tool/workpiece varies between control points the incut segments calculated will not reflect this change. The tool deflection history is not carried over between control points. The initial tool deflection is set to zero for each control point dynamic simulation. Figure 5.14a demonstrates the effect that this will have on the dynamic cutting force. Force peaks 1 to 8 represent a dynamic simulation for tool revolutions 10

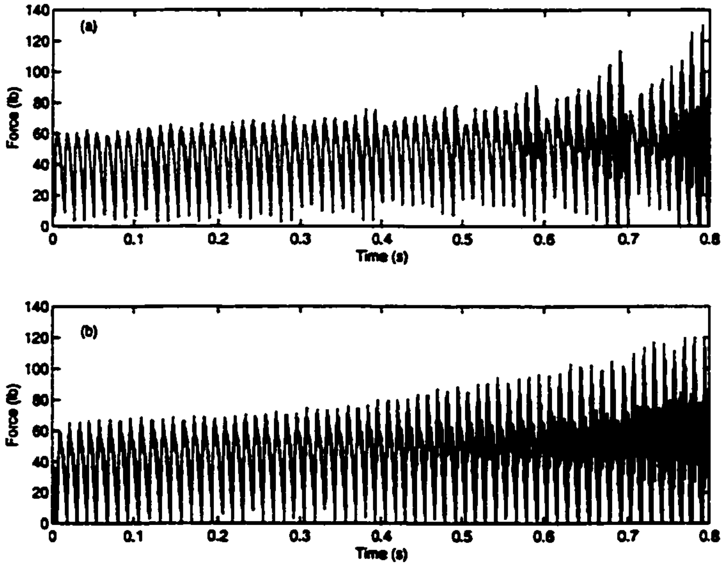


Figure 5.13: Resultant Force - Variable Immersion

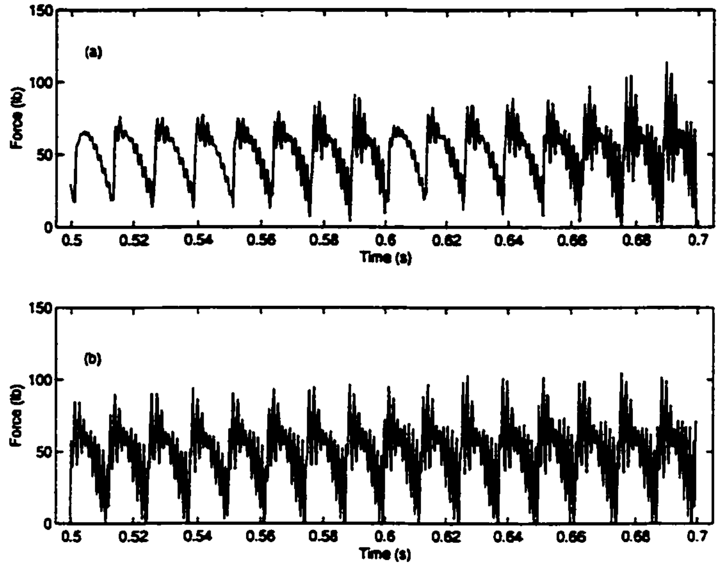


Figure 5.14: Resultant Force (Tool Revolution 10-14)

and 11. Force peaks 9 to 16 represent a dynamic simulation for tool revolutions 12 and 13. The tool deflection is set to zero at force peak 1 and 9. The tool deflection history is then only used for the dynamic simulation over the user specified 2 revolutions of the tool (ie. 1 to 8, 9-16).

Figure 5.13b and 5.14b show the resultant force for the continuous dynamic modeling methodology. Here the incut segments are calculated for the instantaneous location of the tool cutting edge. The instantaneous location of the tool cutting edge is determined from the feed per time step and the current deflection of the tool. The tool deflection history is maintained continuously over the entire tool path. These figures show that the resultant force tracks the variation of the tool/workpiece immersion.

The modeling methodologies described in sections 3.6, 3.7, and 3.8 were compared here for simple linear cuts. The modeling methodologies based on control points offer a piece-wise continous representation of the static and dynamic cutting force and tool deflection over a given tool path. The accuracy of the cutting force prediction and the calculation of tool deflection degrades for variable tool/workpiece immersion. The continous dynamic simulation methodology offers a good representation of the static and dynamic cutting force and tool deflection for variable tool/workpiece immersion.

5.5.2 Sculptured Surface Machining

This section demonstrates the continuous dynamic process simulation methodology for a complex sculptured surface machining operation. A complex sculptured surface part is machined from a titanium block using a 3/4" solid carbide ball end mill (figure 5.15). The feed rate is 10.0 ipm and the spindle speed is 1176 rpm. The machined part

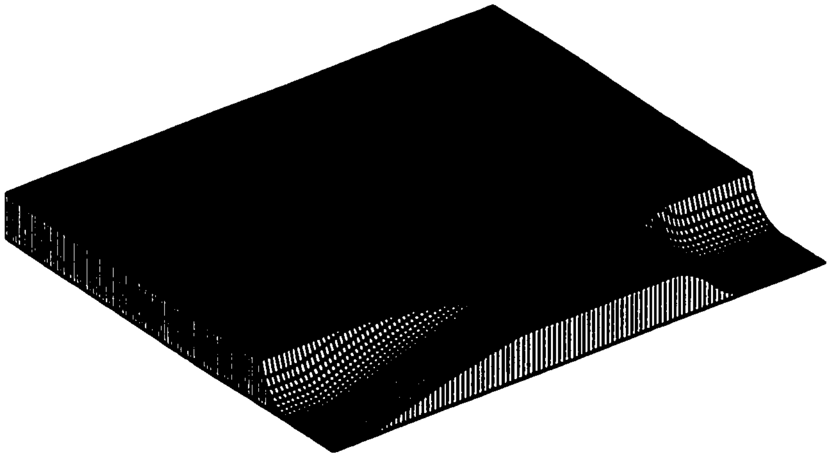


Figure 5.15: AilFoil like Surface - Roughing Operation

was designed to be similar to an airfoil-like surface used in industry. The roughing machining operation consists of five passes on both the leading and trailing edges of the airfoil-like surface. Each set of five passes is repeated at three axial depths of cut on both the leading and trailing edges of the airfoil-like surface. This gives a total of thirty roughing passes.

Simulation results are presented for roughing passes 1 and 2 on the leading edge of the airfoil-like surface. Figure 5.16 shows the resultant cutting force for roughing pass 1 (5.16a) and roughing pass 2 (5.16b). The tool deflection history for each roughing pass is given in figures 5.17 and 5.18. Figure 5.17a shows the tool deflection in the X-direction for roughing pass 1. Figure 5.17b shows the tool deflection in the Y-direction for roughing pass 1. Figure 5.18a shows the tool deflection in the X-direction for roughing pass 2. Figure 5.18b shows the tool deflection in the Y-direction for roughing pass 2.

Each of these figures demonstrates how the continuous dynamic process simulation

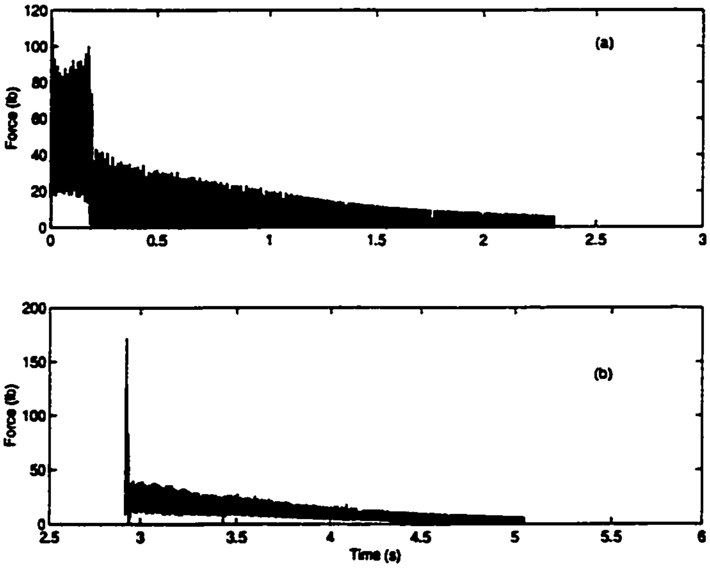


Figure 5.16: Resultant Force - a) Roughing Pass 1 b) Roughing Pass 2

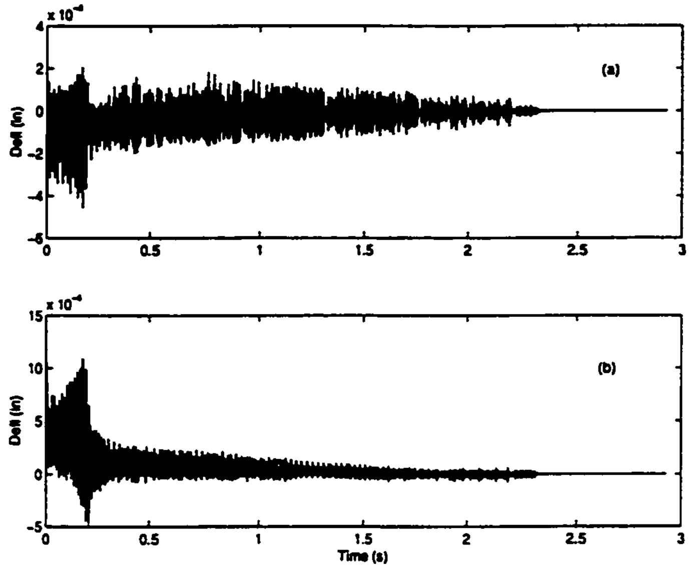


Figure 5.17: Tool Deflection - Roughing Pass 1

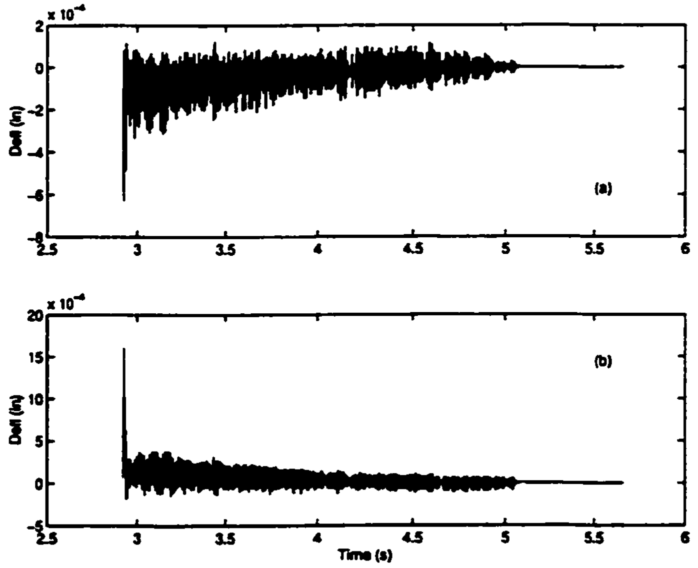


Figure 5.18: Tool Deflection - Roughing Pass 2

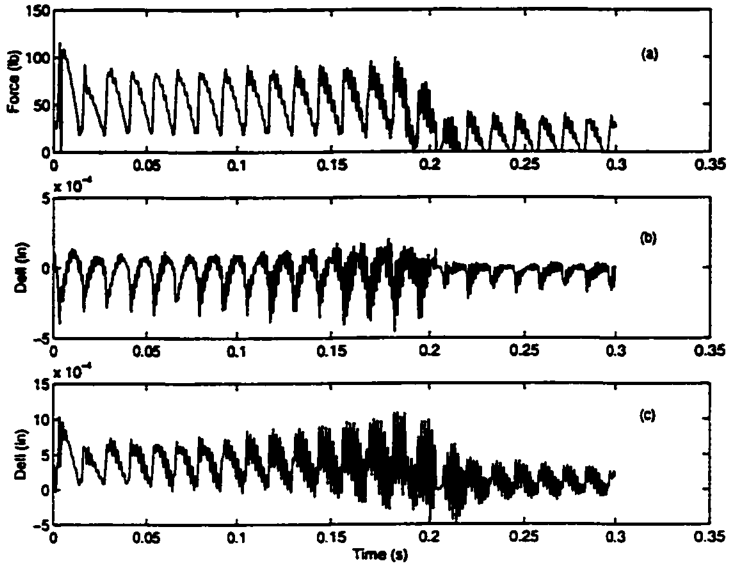


Figure 5.19: Roughing Pass 1 Tool Entry

methodology is capable of predicting the cutting force and tool deflection for variable tool/workpiece immersions that occur during complex sculptured surface machining operations. The ability to predict the cutting force and tool deflection for complex sculptured surface machining allows the user to gain valuable insight into the cutting process. Tool path segments that result in excessive force or tool deflection can be identified from the simulation results. As an example, the tool entrance into the workpiece for roughing pass 1 is shown in figure 5.19. Figure 5.19a shows the resultant cutting force. Figure 5.19b and 5.19c show the tool deflection in the X and Y directions respectively. The tool/workpiece immersion at the start of roughing pass 1 is high. This results in large cutting forces and tool deflections. Large cutting forces and tool deflections may cause tooth chipping or breakage. Having identified this tool path segment as potentially problematic, the user may decide to specify a lower entrance feed rate or alter the tool path specification for this segment. The goal is to reduce the cutting force and tool deflection. The continuous dynamic process simulation can be used to verify that the changes made will result in the desired cutting force and tool deflection.

5.6 Internet Based Machining System Applications

The Internet-based facility for multi-axis milling process simulation realizes an Internet enabled process optimization environment. This environment leads the manufacturing engineer through formalized procedures in which physics based process simulations allow process parameters and operational sequences to be evaluated and

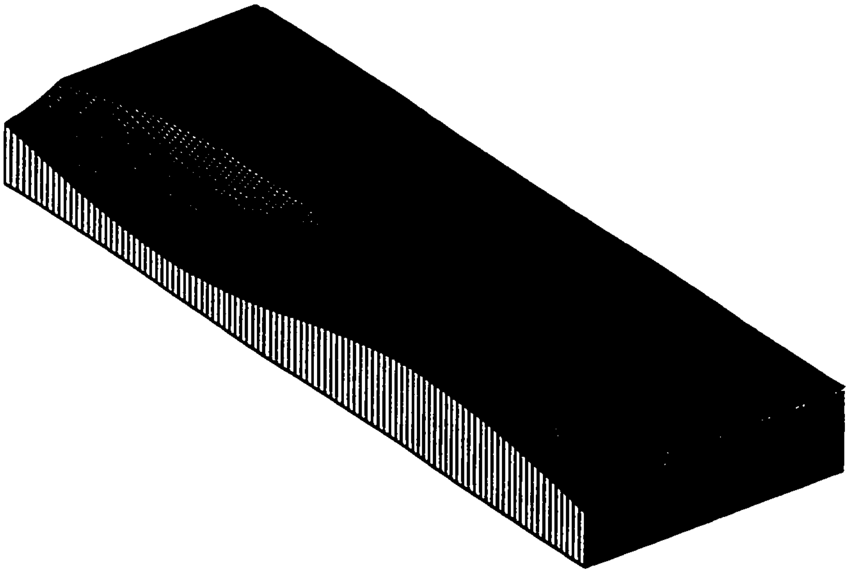


Figure 5.20: Simple Workpiece Geometry

optimized with respect to desired product features. To illustrate some of the possible applications for the Internet-based multi-axis milling process simulation, cutting force and tool deflection are considered in section 5.6.1.

5.6.1 Cutting Force and Tool Deflection

Predicting the cutting force and tool deflection offers the manufacturing engineer valuable insight into the process. This information can be used for reducing or eliminating the need for expensive process trials. Specific process problems can also be diagnosed more efficiently when this information is available. Having this information allows the engineer to optimize the manufacturing process by testing different process parameters (feed, speed, tool paths, etc.).

The process information is presented to the manufacturing engineer in various ways. The cutting force and tool deflections can be represented as instantaneous values at specific tool path locations or can be represented globally over the entire tool path.

The simple workpiece geometry shown in figure 5.20 is used to illustrate both cutting force and tool deflection. To perform a simulation, the user must input parameters related to the workpiece, cutting tool, and tool path. The workpiece input parameters are the stock dimensions and material type (titanium Ti6Al4V). The cutting tool input parameters are the tool type (ball), diameter (3/4 in.) and material (solid carbide). The tool path input parameter is the name of the CLData file. The cutting speed and feed per tooth used for these simulation were 215 sfpm and 0.003 inch.

The cutting force for the simple workpiece geometry are presented in figures 5.21 and 5.22. Figure 5.21 shows the global representation of the cutting force over the entire tool path. Figure 5.21a shows the predicted cutting force using the simulation. Figure 5.21b shows the cutting force obtained from an actual cutting experiment. In Figure 5.21a the value of each point on the force curve is equal to the maximum cutting force at the corresponding tool path position. Two reference locations (Pt 1, Pt 2) are labeled in figure 5.21a. The instantaneous resultant cutting force versus tool rotational angle is shown in figure 5.22 for each reference location. Figure 5.22a shows the resultant cutting force for reference Pt 1. Figure 5.22b shows the resultant cutting force for reference Pt 2. The value of the cutting force at each reference location, given in figure 5.21, is equal to the maximum instantaneous resultant cutting force given in figure 5.22.

The tool deflections along the tool path for the simple workpiece geometry are

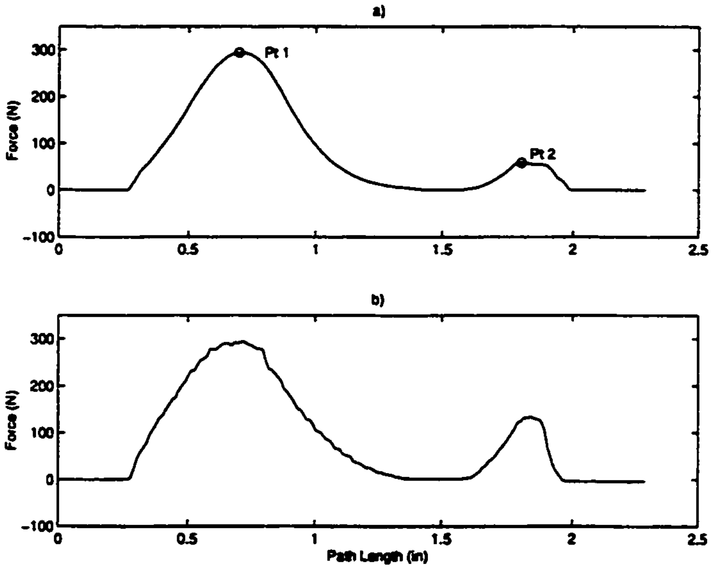


Figure 5.21: Maximum Cutting Force Over Entire Tool Path, a) Predicted, b) Experimental

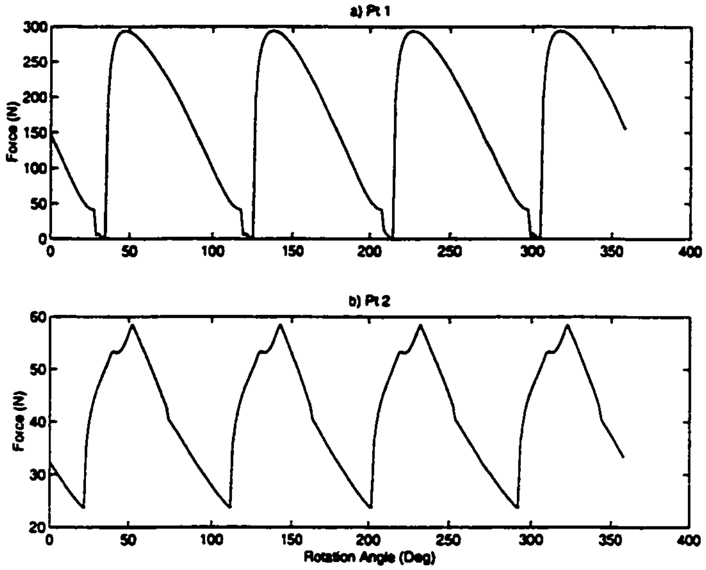


Figure 5.22: Inst. Cutting Force a) Pt 1, b) Pt 2

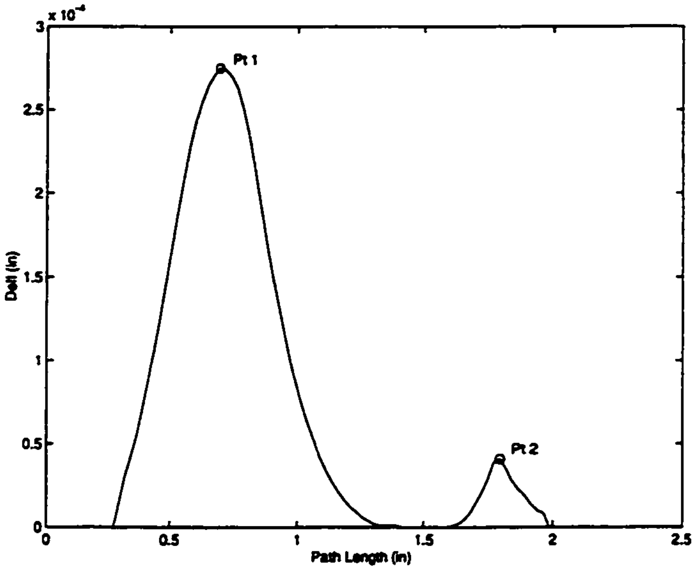


Figure 5.23: Maximum Tool Deflection Over Entire Tool Path

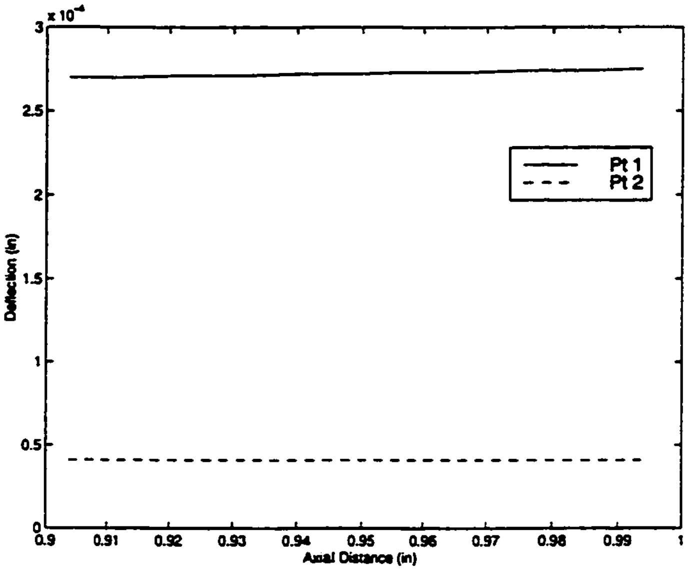


Figure 5.24: Inst. Tool Deflection a) Pt 1, b) Pt 2

presented in figures 5.23 and 5.24. The tool deflections shown in these figures are normal to the path direction. Figure 5.23 shows the global maximum tool deflection over the entire tool path. The value of each point on this curve is equal to the maximum tool deflection at the corresponding tool path position. The same two reference locations (Pt 1, Pt 2) are labeled on this figure. The tool deflection, expressed as a function of the axial distance from the tool holder, is shown in figure 5.24 for each reference location. The value of the tool deflection at each reference location, given in figure 5.23, is equal to the maximum tool deflection given in figure 5.24.

5.7 Summary

This chapter provided an overview of the simulations and experimental verification for the modeling methodologies presented in chapter 3. Experimental results were presented for the calibration procedure. The cutting force calibration coefficients account for the effects of feed rate, cutting speed, and a complex cutting edge design. Model verification tests were conducted with these cutting force calibration coefficients. These tests demonstrate that the predicted forces are within 5% of experimentally measured forces. An overview of some of the simulation capabilities was discussed. Simulated results were shown for predicting dynamic cutting forces and static/dynamic tool deflection.

The enhanced approach for dynamic mechanistic modeling methodology for multi-axis machining was demonstrated. The continuous dynamic process simulation methodology was compared with both static and dynamic simulation methodologies based on control points. The comparison was for linear cuts with constant and variable tool/workpiece immersion. This comparison showed that the modeling methodologies based on control points only offer a piece-wise continous representation of the static and dynamic cutting force and tool deflection over a given tool path. The accuracy of the cutting force prediction and the calculation of tool deflection using methodologies based on control points degrades for variable tool/workpiece immersion. The continous dynamic simulation methodology demonstrates a good representation of the static and dynamic cutting force and tool deflection for variable tool/workpiece immersion. The continous dynamic process simulation methodology was demonstrated for a complex sculptured surface machining operation. The rough machining operation for an airfoil-like surface was presented. The simulation results demonstrate how the continuous dynamic process simulation methodology is capable of predicting the cutting force and tool deflection for variable tool/workpiece immersions that occur during complex sculptured surface machining operations.

Applications of the Internet based machining system were discussed. Simulation results were presented for a simple sculptured surface part. Both cutting force and tool deflection were given. The cutting force and tool deflection can be presented to the user as either instantaneous values or globally over the entire tool path. This information gives the user valuable insight into the machining process.

Chapter 6

Feed Scheduling Results

6.1 Introduction

During machining of complex sculptured surfaces, the chip geometry is constantly changing along the tool path. This causes wide fluctuations in the process variables (cutting force and tool deflection). In the worst case, the tool may break or cutting may become unstable (chatter) and/or the quality of the finished product may be unacceptable. To ensure stable machining conditions and guarantee the quality of the finished product, conservative process parameters are typically used (feeds and speeds). The result is lower productivity. Feed rate scheduling can be used to optimize the manufacture of complex sculptured surface parts. The productivity will be increased with guaranteed machining integrity.

This chapter demonstrates the feed scheduling concept for the methodologies presented in chapter 4. Section 6.2 presents a case study for optimizing the machining process for a complex sculptured surface part. Section 6.3 demonstrates the methodologies for incorporating the feed drive performance capabilites with feed scheduling.

6.2 Complex Sculptured Surface Part

A case study of manufacturing a complex free form surface produced by a 3-axis milling process is used to demonstrate the feed scheduling methodology for multi-axis machining presented in section 4.2. A complex sculptured surface part was machined from a titanium block using a 3/4 inch solid carbide ball end mill (figure 6.1). The machined part was designed to be similar to an airfoil-like surface used in industry. The roughing machining operation consisted of five passes on both the leading and trailing edges of the airfoil-like surface. Each set of five passes was repeated at three axial depths of cut on both the leading and trailing edges of the airfoil-like surface. This gave a total of thirty roughing passes. Figure 6.2 shows intermediate part

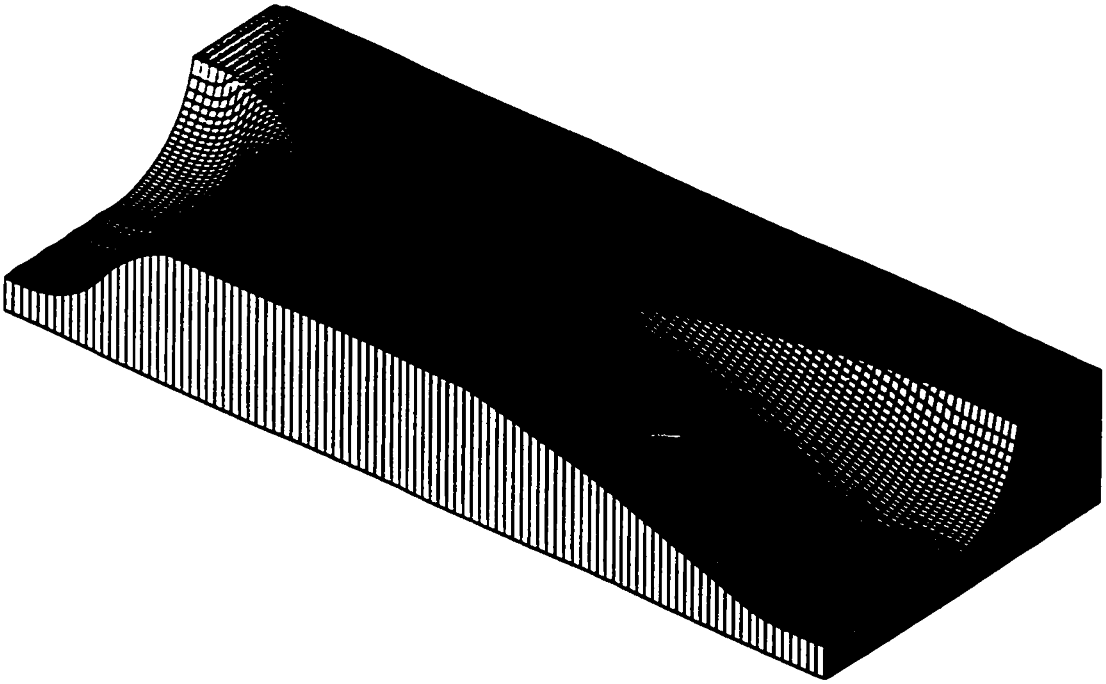


Figure 6.1: Airfoil Like Surface Section

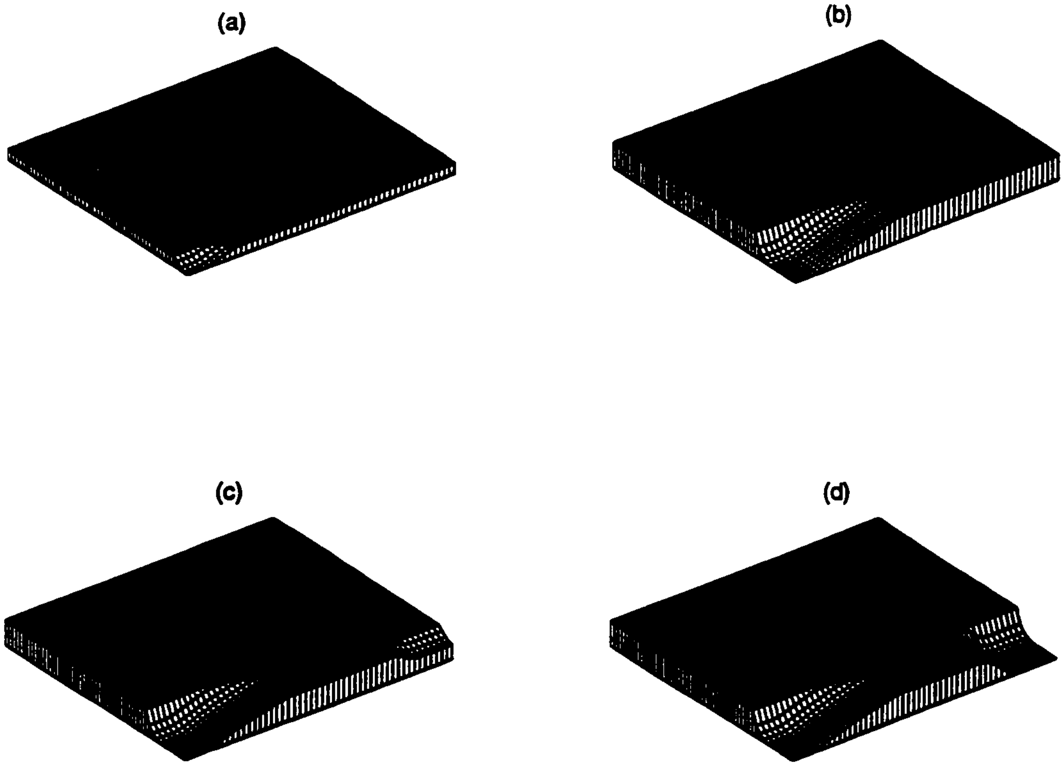


Figure 6.2: AirFoil like Surface Roughing Stages

geometry for the roughing operation generated by the modeling procedure detailed in section 3.2. Figure 6.2a is the part geometry after the first roughing pass on the trailing edge of the airfoil-like surface. Figure 6.2b is the part geometry after the fifteen roughing passes on the trailing edge of the airfoil-like surface. Figure 6.2c is the part geometry after the first roughing pass on the leading edge of the airfoil-like surface. Figure 6.2d is the part geometry after the fifteen roughing passes on the leading edge of the airfoil-like surface.

Feed rate scheduling is performed for both maximum chipload and maximum force constraints. Results of the different simulation runs are shown in figures 6.3 and 6.4.

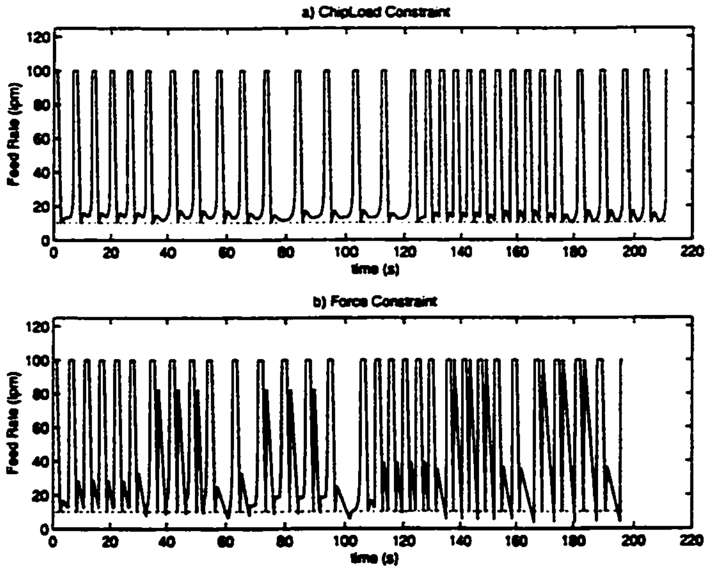


Figure 6.3: Scheduled Feed Rates

Figure 6.3 shows the scheduled feed rates. Figure 6.4 shows the predicted force. The force profile shown in figure 6.4 is the maximum force at each simulation time step. Each force spike represents a single roughing pass. There are a total of 30 roughing passes. Figure 6.4a is the force profile for feed scheduling using a maximum chipload constraint. Figure 6.4b is the force profile for feed scheduling using a maximum force constraint. Figure 6.4c is the force profile for a nominal feed of 0.002 inch per tooth. The production savings is evident through a comparison of the total machining times in each figure. The time to rough machine the part using a nominal feed is 295 seconds (fig. 6.4c). The time to rough machine the part with feed scheduling using a chipload constraint is 210 seconds (fig. 6.4a) and the machining time using a force constraint is 195 seconds (fig. 6.4b). By using feed scheduling with a chipload constraint, the machining time is reduced by 28.7%. Feed scheduling with a force constraint reduced the machining time by 33.7%. Figure 6.4b also demonstrates how feed scheduling can be used to help guarantee machining integrity. The force is regulated at a level of 450N. Force regulation is a method that can be used to guarantee machining integrity.

Experimental tests were performed to demonstrate the production savings attained using feed scheduling with a maximum chipload constraint. Figure 6.5a shows the resultant cutting force versus time for the roughing operation with no feed scheduling. Figure 6.5b shows the resultant cutting force versus time for the roughing operation with feed scheduling using the maximum chipload constraint. The production savings is evident through a comparison of the total time required to machine the airfoil-like section in each case. With no feed scheduling, the total time of machining was 295 seconds. Using feed scheduling, the machining time was 210 seconds. This results in a production savings of 28.7%. Figures 6.6a and 6.6b show the actual airfoil like surface section that was machined with feed scheduling and without feed scheduling respectively.

6.3 Incorporating Feed Drive Performance Capabilities

This section demonstrates the methodologies for incorporating the feed drive performance capabilites with feed scheduling. This section begins with a description of the experimental setup. The results for a feed scheduling strategy based on control points without incorporating feed drive performance capabilities are then given in

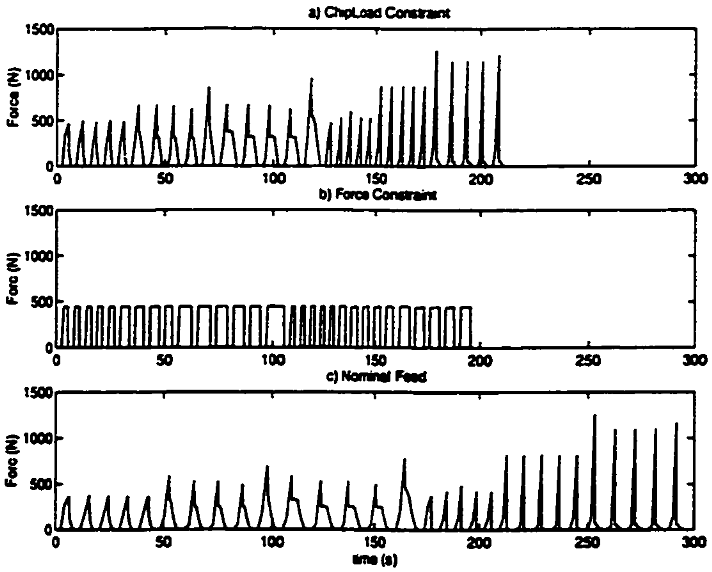


Figure 6.4: Predicted Forces

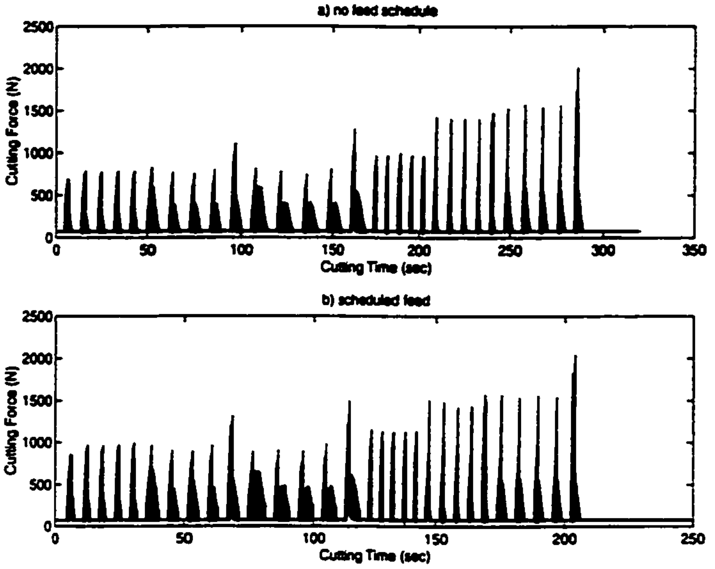
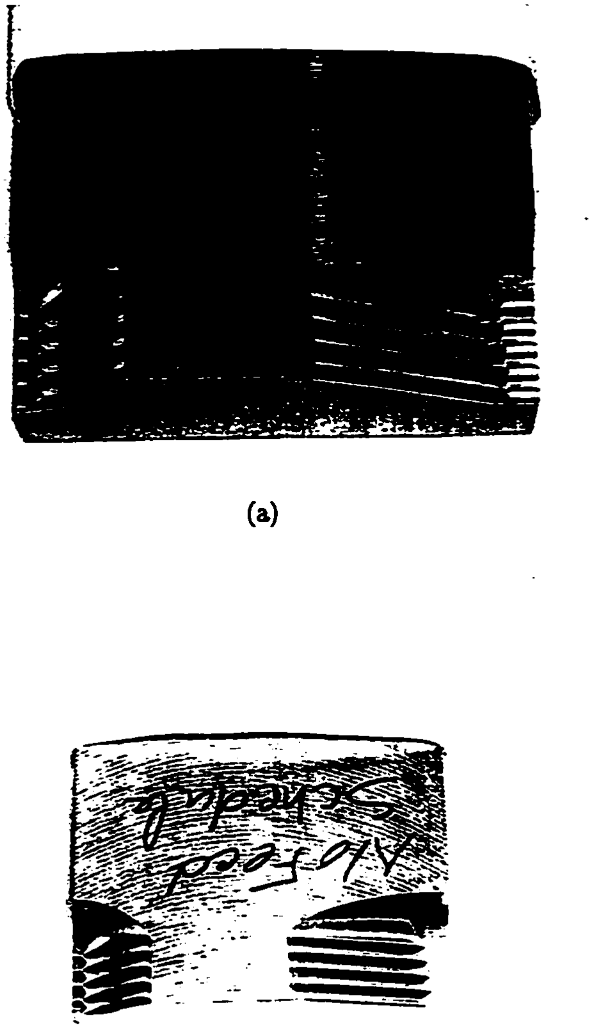


Figure 6.5: Experimental Cutting Forces



(a)

(b)

Figure 6.6: Airfoil Like Surface Section a) Feed Scheduling, b) No Feed Scheduling

6.3.2. The results for the feed scheduling strategy based on the Acc/Dec Time Constant are presented in 6.3.3. The results of the optimized feed scheduling strategy are given in 6.3.4. A comparative summary of all the feed scheduling strategies is given in section 6.3.5.

6.3.1 Experimental Setup

The experiments are carried out on a 5-axis vertical milling machine "FADAL VMC4020".

The G-Code file is generated by the simulation on a personal computer (PC), and then transferred to the CNC control unit via a serial connection RS-232. Each G-Code file is composed of three parts; the header, the body and the footer. The header specifies the general set up of the machine. In this study, the header includes the interpolation plane (XY plane (G17)), the interpolation mode (linear (G01)), the position unit (inch (G20)), the reference mode (incremental (G91)), and the blend mode (with no return to zero feed rate at each change of feed rate (G08)). The function (G08) ensures a continuity of the velocity profile and a higher productivity while machining. The function (M49) is introduced to disable the feed/speed operator override. In addition, two functions, (M60) and (M62), are introduced to actuate pneumatic brakes for the two rotary axis of motion. The function (M3) specifies that the spindle is rotating in a clockwise direction. The function (M8) turns on the coolant. To synchronise the data acquisition with the start of execution of the signal command, the following two M functions, (M67) and (M66), are introduced at the end of the header. The descending slope of the (M67)/(M66) signal is used as an external trigger for the data acquisition system. Thus, the acquisition of all data will start at the same time as the processing of the trajectory blocks. The body is composed of the trajectory specifications. The footer specifies the return of the machine to the home position and the end of the program.

A Kistler milling dynamometer type "9255B" is mounted on the table of the machine to measure the cutting forces during machining. The three axis cutting force signals are sampled at a frequency of 3600 Hz. The forces are recorded using a data translation "DT2801" analog-to-digital acquisition card (ADC). The amplifier and the tachometer signals are supplied to the PC through a separate ADC card. The signals are sampled at a sampling frequency of 100 Hz. Both data acquisition systems are triggered by the signal delivered by the (M67)/(M66) functions. The spindle speed is fixed at 1200rpm. The tool used is a 330M Garr 3/4" diameter, solid carbide ball end mill. The ball end mill has 4 helical flutes.

The workpiece material is Titanium (Ti6Al4V). To evaluate the performance of the different feed scheduling strategies, successive channels modulate the surface of the workpiece (figure 6.7a). A cross section of the workpiece is depicted in figure 6.7b. This surface offers various changes in the axial depth of cut: constant, sudden or smooth decrease, and sudden or smooth increase in the depth of cut. The radial depth of cut is kept constant during each test. It is equal to 1/3 of the cutter radius. This workpiece is used for all experiments and as a reference for all simulations performed. Between two successive ball end milling experiments, a flat end milling operation is performed to regenerate the workpiece edge.

To demonstrate the productivity saving, two sets of experiments are conducted. In the first set, the tool path is short so that the air cutting-to-cutting ratio is small. In the second set of experiments, the tool path is long providing a higher air cutting-to-cutting ratio (figure 6.8).

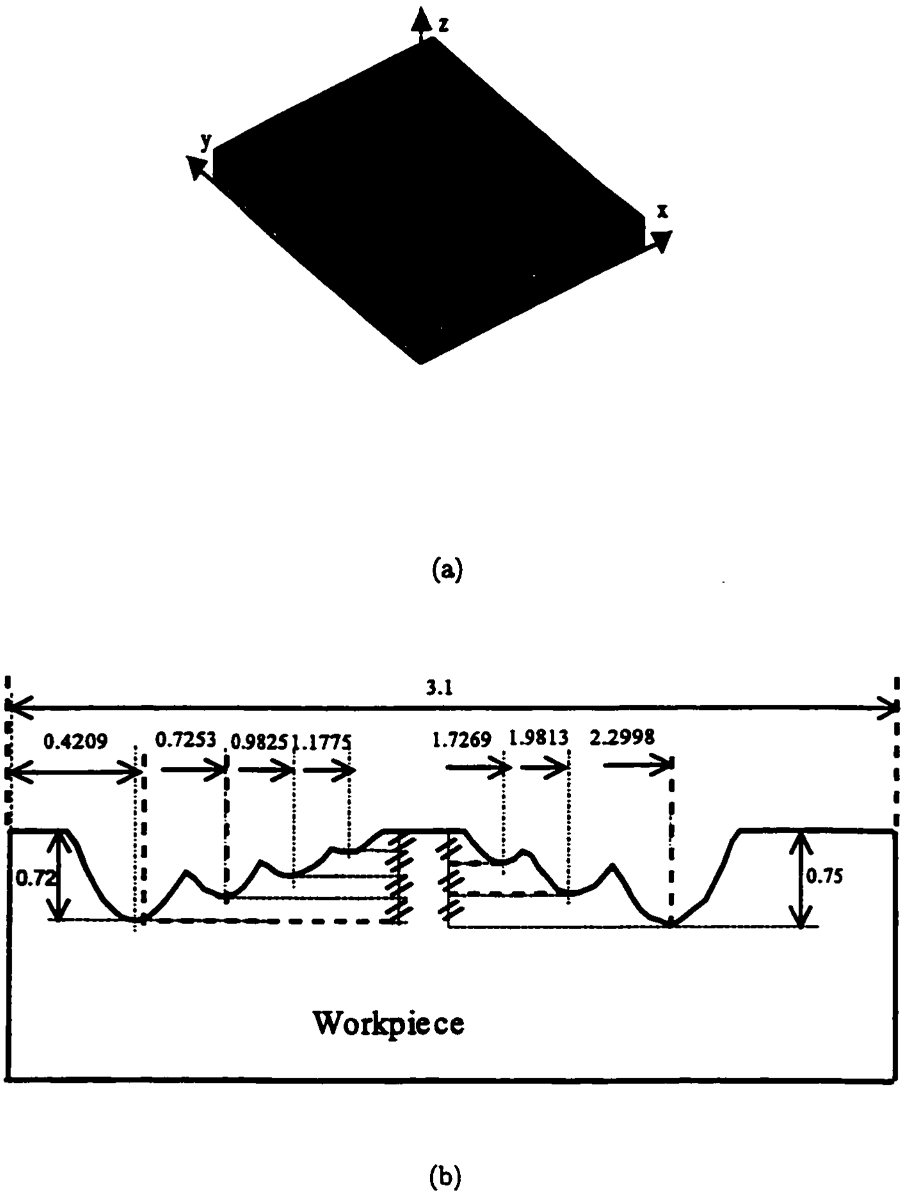


Figure 6.7: Sketch of the workpiece

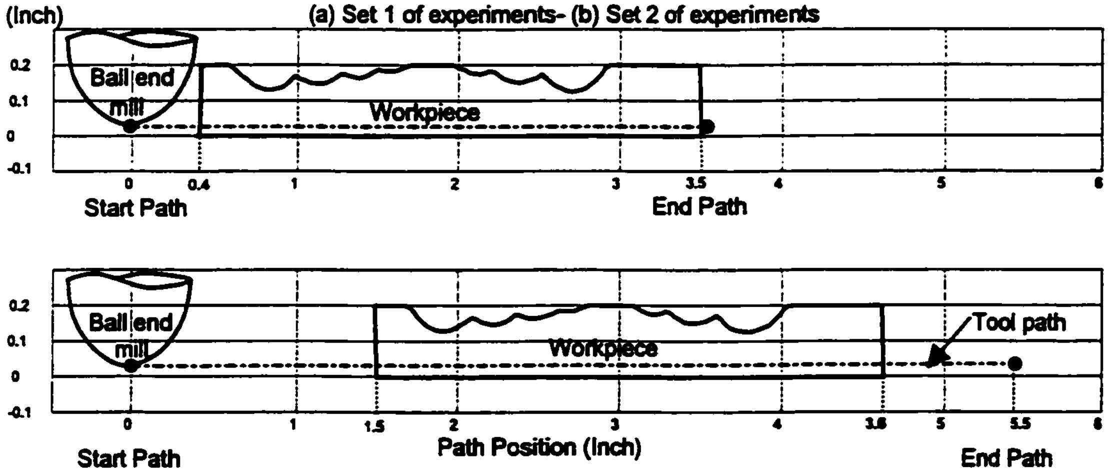


Figure 6.8: Tool paths for the two sets of experiments

Calibration of the Force Model

End milling operations are performed with a constant feed rate, $F_n = 28.8ipm$, on the specified workpiece for the two sets of experiments. These experimental results are used to calibrate the mechanistic force model. The results are also used to provide the reference time and target force values in order to demonstrate the effectiveness of the feed scheduling as well as the usefulness of incorporating the feed drive dynamics.

In the region where the cutting conditions are constant (i.e., where the workpiece is flat), a cutting force profile for a single tooth period is extracted and the calibration procedure given in section 3.4 is applied. The force coefficients resulting from this calibration procedure are given in table 6.1:

Figure 6.9 shows the three cutting force components as well as the resultant force profile for a single tooth period. The thin lines represent the profiles of the averaged force based on the experimental data while the thick lines represent the profiles of

Rake face			Clearance Face	
K_1 ($N/inch^2$)	K_2 ($N/inch^2$)	K_3 ($N/inch^2$)	K_4 ($N/inch$)	K_5 ($N/inch$)
676913.48	888472.77	16059.56	173.73	116.46

Table 6.1: Force coefficients

simulated force using the calibration constants. In all three components of the force there is a very close agreement in both magnitude and shape.

To validate the calibration procedure, the average cutting force obtained from the cutting experiment is compared to the force obtained from the simulation. An average cutting force is defined as follows. For each revolution of the tool, the maximum cutting force magnitudes of each tooth period are averaged. Repeating this for each revolution of the tool along the tool path results in the average cutting force. The average cutting force only contains the average peak force for each tool rotation, resulting in a force trace that has no variations due to the cutter run-out or slight changes of cut geometry (figure 6.10).

In the time domain, the simulated force (thick line) is shown with the experimental data (thin line) in figure 6.11a. The shapes of the two plots agree well. An expansion in time is noticed in the experimental cutting force compared to the simulated cutting force. This expansion in time increases with the travel period. The estimated machining period is equal to 7.25s. However, the actual machining period is equal to 7.67s. The noticeable expansion in time is due to the following reasons. The simulation is run under ideal conditions of motions. It does not take into account the friction of the feed drive mechanism. The friction acts to increase the period of the travel. In addition, the feed rate jump is achieved instantaneously in the simulation. The actual system needs a period of time to accelerate or decelerate. This will

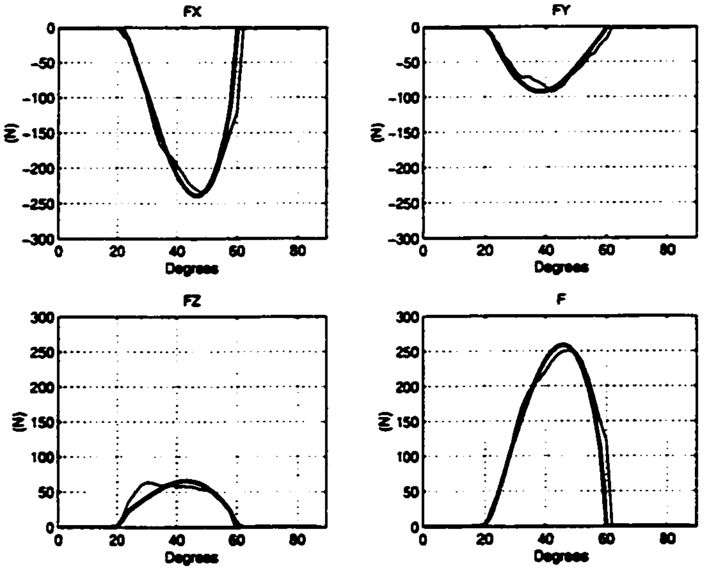


Figure 6.9: Mechanistic force model calibration

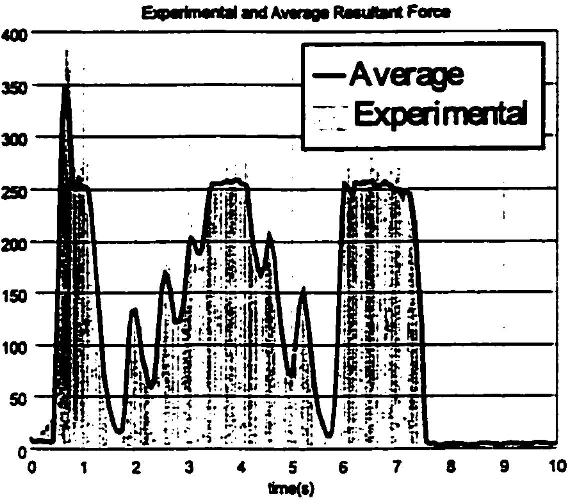
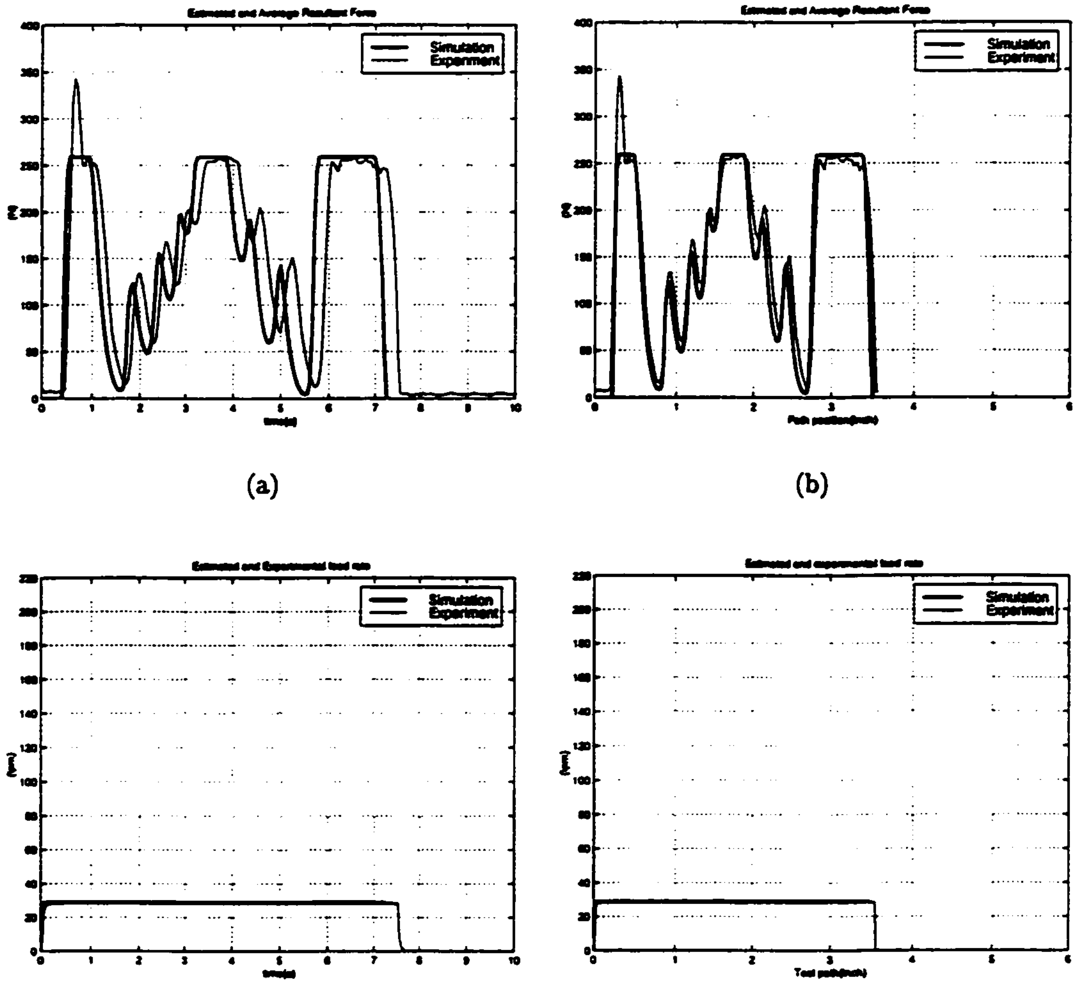


Figure 6.10: Average cutting force concept



(a) (b)

(c) (d)

Figure 6.11: Mechanistic force model validation,(a) Cutting force: Time representation, (b) Cutting force: Spatial representation, (c) Feed rate: Time representation, (d) Feed rate: Spatial representation.

be demonstrated in the following experiments. A spatial presentation of the results is proposed. The x-axis is the tool path position. At an instant t, $t = k \cdot T$, x is obtained by integrating the experimental/simulated feed rate with respect to time:

$$x = \int_0^t F(t) \cdot dt \cong \sum_{i=0}^{k} F(i \cdot T) \cdot T \tag{6.1}$$

Good agreement between experimental and simulated results is observed when using the spatial representation (figure 6.11b). The calibration constants are accepted and are used for further development and analysis.

6.3.2 Control Point Strategy

End milling operations are carried out for the two sets of experiments. The predicted and experimental cutting force are depicted in both the time and spatial domains in figures 6.12a, 6.12b, 6.13a, and 6.13b. The actual feed rates generated by the simulation are also plotted against the experimental feed rate in both the time and spatial domains in figures 6.12c, 6.12d, 6.13c, and 6.13d.

Figures 6.12d and 6.13d clearly show that the acceleration limitations of the feed drive do not allow the predicted feed rates to be reproduced accurately. The target feed rate is undershot when approaching the workpiece, as well as at the end of the path for the first set of experiments (figure 6.12d). The feed drive system also undershoots the target feed rate at many locations along the tool path for both sets of experiments. Consequently, the target cutting force is undershot. Although undershooting the target cutting force gives poor regulation of the force, the machining process is still safe. However, overshooting the target feed rate at some locations along the tool path results in the target force being exceeded. If the target cutting force is exceeded, the tool may break.

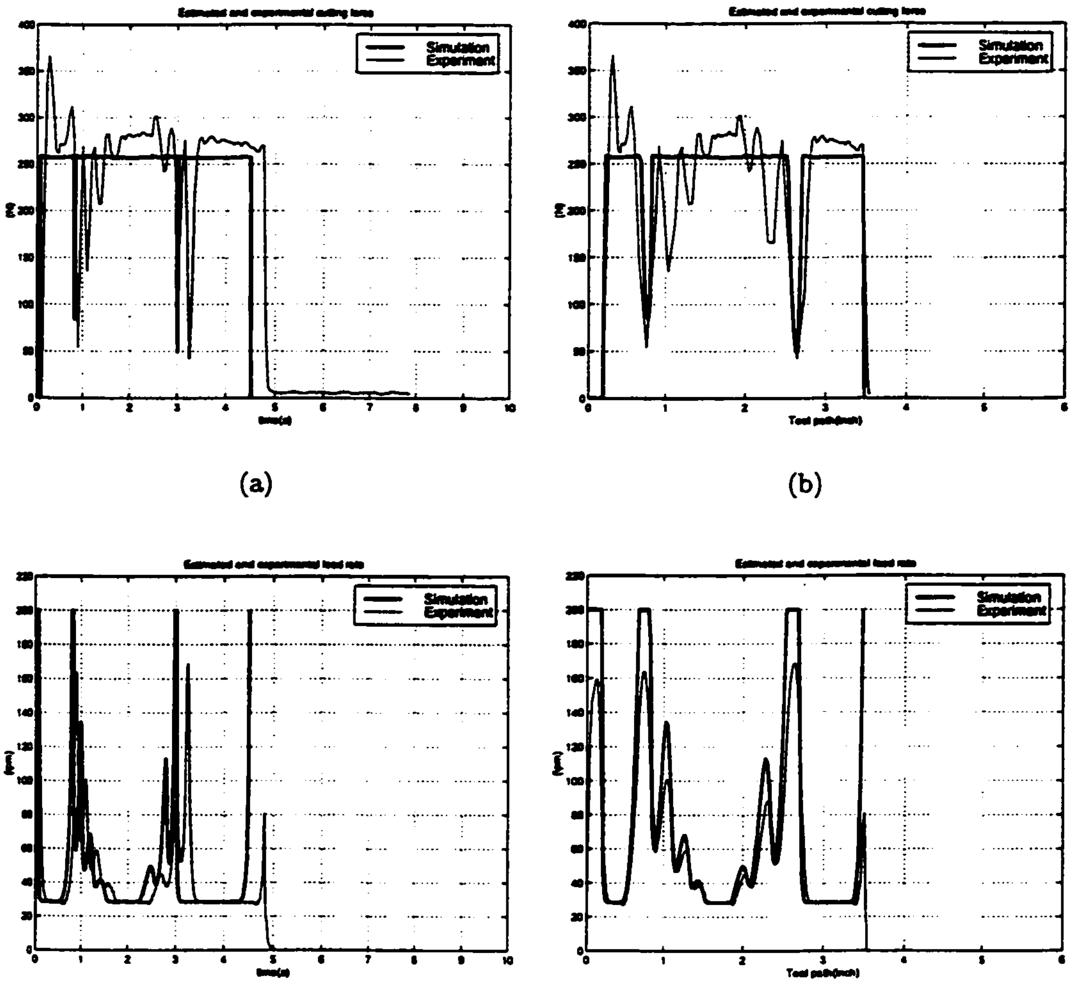
In all the plots represented in the time domain (figures 6.12a, 6.12c, 6.13a, and 6.13c) a time shift between the predicted and experimental results is observed. This shift increases with the travel period and is explained by the presence of the friction in the feed drive system. The inability of the actual feed drive system to achieve the target feed rate results in an increase of the travel period as well.

The estimated and experimental machining time for the short tool path length are equal to 4.50s and 4.99s, respectively. For the long tool path length, the estimated and experimental machining time are equal to 5.10s and 5.67s, respectively. Applying the feed scheduling strategy using control points improves the productivity for these tests by 35% and 51%, respectively. This strategy fails to achieve the target feed rate at the appropriate position, resulting in poor regulation of the force and risks of unsafe machining due to the tool breakage.

6.3.3 Acc/Dec Time Constant Strategy

End milling operations are carried out for the two sets of experiments. The predicted and experimental cutting force are depicted in both the time and spatial domains in figures 6.14a, 6.14b, 6.15a, and 6.15b. The actual feed rates generated by the simulation are also plotted against the experimental feed rate in both time and spatial domains in figures 6.14c, 6.14d, 6.15c, and 6.15d.

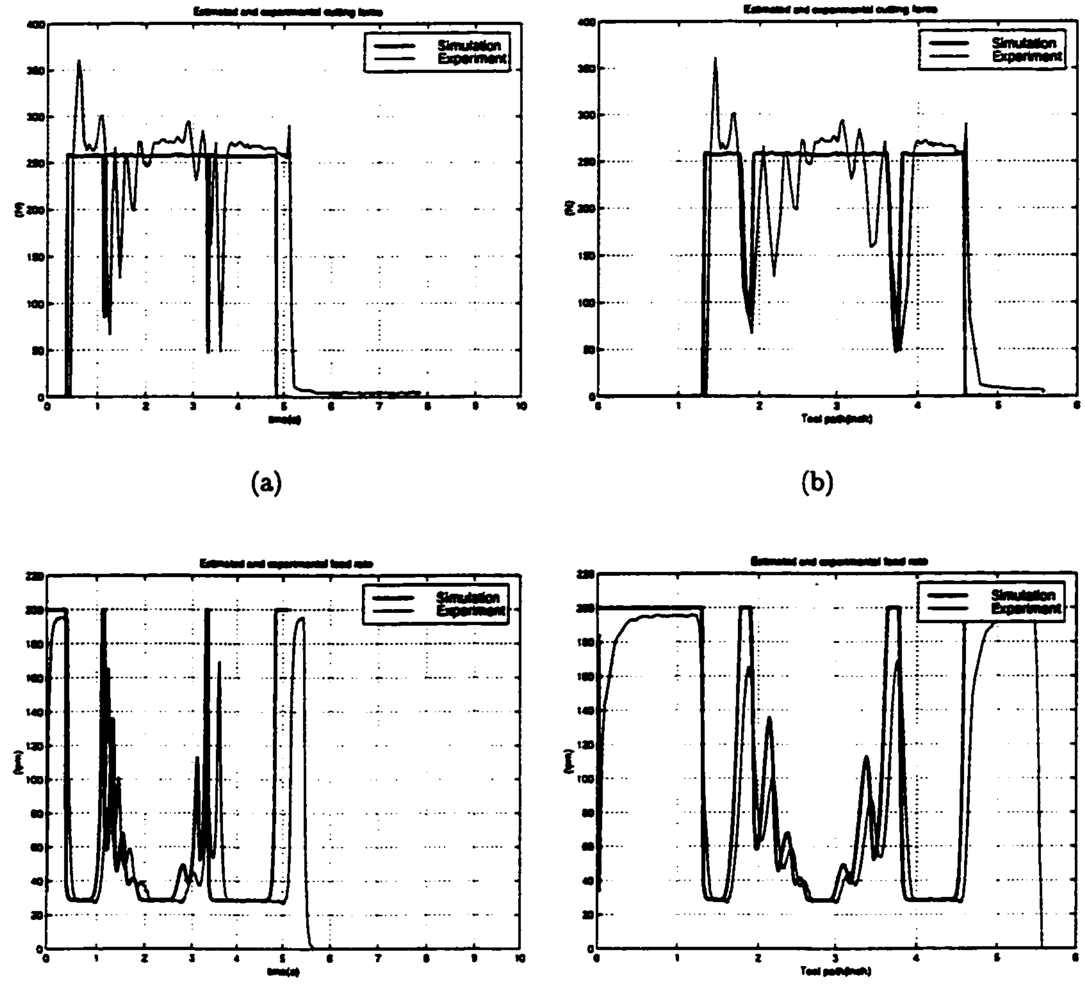
In the time domain, experimental and predicted data have the same profile but a time shift is evident. The time shift is caused by the friction in the feed drive system. In the spatial domain, good agreement is observed between predicted and experimen-



(a) (b)

(c) (d)

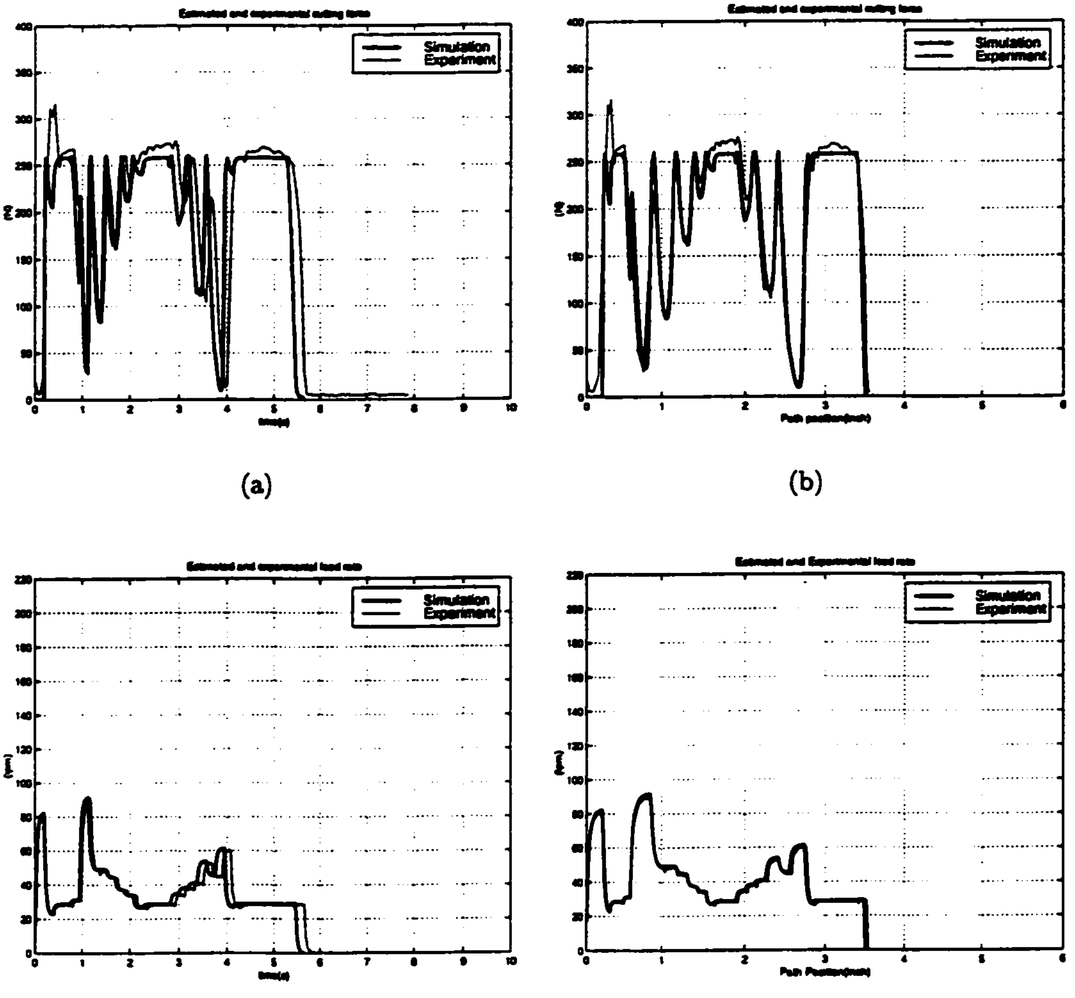
Figure 6.12: Feed scheduling strategy based on control points, short tool path,(a) Cutting force: Time representation, (b) Cutting force: Spatial representation, (c) Feed rate: Time representation, (d) Feed rate: Spatial representation.



(a) (b)

(c) (d)

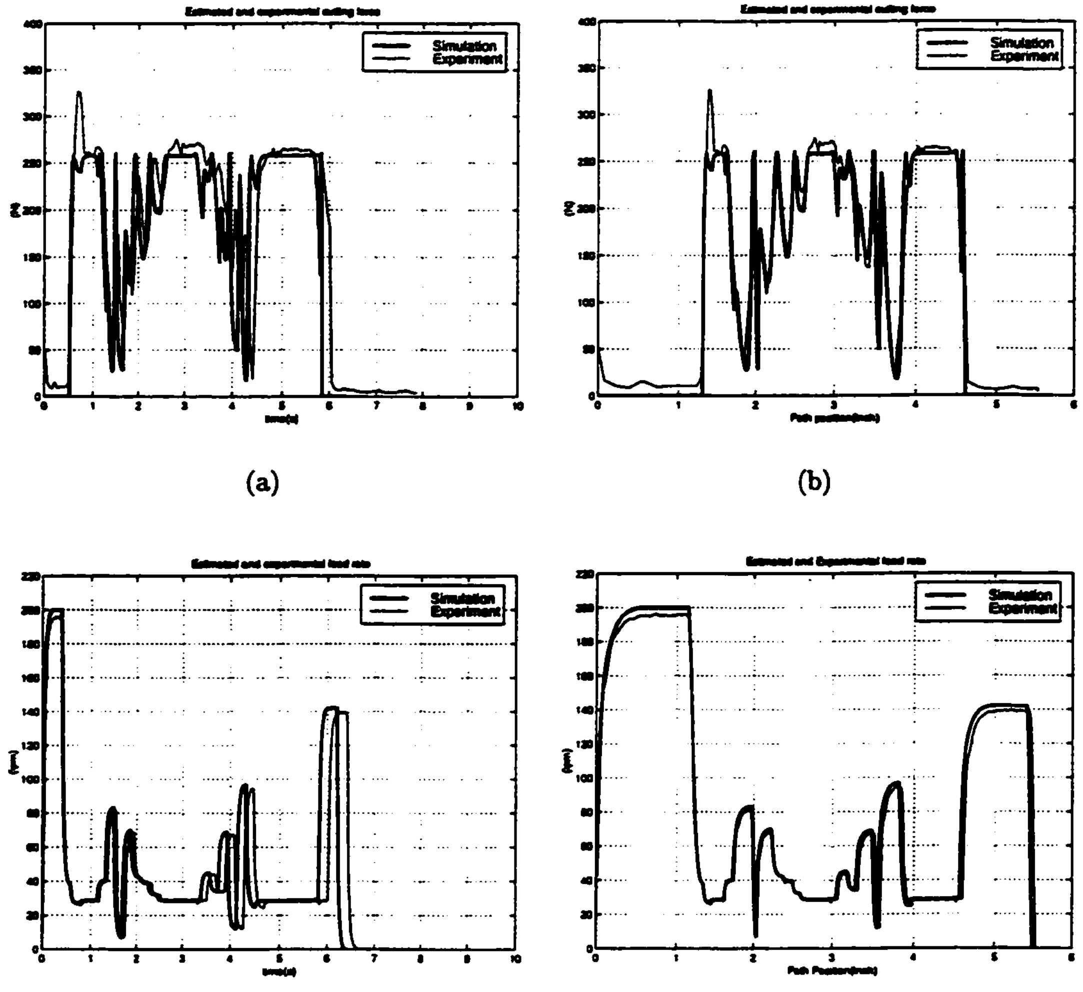
Figure 6.13: Feed scheduling strategy based on control points, long tool path,(a) Cutting force: Time representation, (b) Cutting force: Spatial representation, (c) Feed rate: Time representation, (d) Feed rate: Spatial representation.



(a) (b)

(c) (d)

Figure 6.14: Feed scheduling strategy based on Acc/Dec time constant, short tool path,(a) Cutting force: Time representation, (b) Cutting force: Spatial representation, (c) Feed rate: Time representation, (d) Feed rate: Spatial representation.



(a) (b)

(c) (d)

Figure 6.15: Feed scheduling strategy based on Acc/Dec time constant, long tool path,(a) Cutting force: Time representation, (b) Cutting force: Spatial representation, (c) Feed rate: Time representation, (d) Feed rate: Spatial representation.

tal actual feed rate and cutting force. By integrating the feed drive dynamics with feed scheduling, the target feed rate is achieved at the appropriate position. The plots of the cutting force clearly show variations in the force magnitude. Although this magnitude is less than the target force value and there is no risk of the tool breakage, the force regulation is poor. This is explained by considering the constraint imposed on the actual feed rate. The actual feed rate has to reach the commanded feed rate. The feed scheduling interval is set equal to the Acc/Dec time constant of the machine. This means that changes in the feed rate may be slower than changes in the cutting geometry resulting in a poor regulation of the force. This constraint is very conservative, as the feed drive system is capable of changing the feed rate more rapidly than this strategy demands.

The estimated and experimental machining time, for the short tool path length, are equal to 5.66s and 5.82s, respectively. For the long tool path length, the estimated and experimental machining times are equal to 6.38s and 6.65s, respectively. Applying the feed scheduling strategy based on the Acc/Dec time constant has improved the productivity for these tests by 24% and 42%, respectively. Although this strategy allows an accurate prediction of the feed rate and the cutting force, it does not provide good force regulation.

6.3.4 Optimized Strategy

End milling operations are carried out for the two sets of experiments. For each set of experiments (short tool path and long tool path), the G-Code file is generated using the optimized feed scheduling strategy. The experimental cutting forces (thin line) are plotted against the predicted cutting force (thick line), in both time and spatial domain in figures 6.16a, 6.16b, 6.17a, and 6.17b. The experimental feed rate (thin line) is plotted with the predicted feed rate (thick line) in both the time and spatial domains in figures 6.16c, 6.16d, 6.17c, and 6.17d. From the time domain representations (figures 6.16a, 6.16c, 6.17a, and 6.17c) a shift in time is observed between experimental and predicted signals. This shift is due mainly to the friction effects of the feed drive system and an error in the predicted gain of the transfer function for the feed drive system. The friction acts to slow down the motion, increasing the actual time of travel. The actual time for the tool motion is greater than that predicted. Friction is a nonlinear effect, and is not modeled in this study. The effects of friction are observed in figure D.8 in appendix D. The experimental Acc/Dec profiles derived from the tachometer signal for different feed rate jumps (thin lines) exhibit high frequency components.

The transfer function of the feed drive system is identified in appendix D along with the transfer function of the disturbances. The disturbances introduce an error in the gain of the transfer function of the feed drive system. This error is less than 1.5%. The transfer function of the disturbances is not integrated with the optimized feed scheduling strategy. The error in the gain of the transfer function is manifested mainly during tool motions with constant feed rate (figure 6.17d).

Based on the spatial representation (figures 6.16b, 6.16d, 6.17b, and 6.17d), good agreement is observed between experimental and predicted feed rate. An error of 1.5% is evident in the feed rate magnitude. The predicted feed rate reproduces the following accurately:

- The acceleration of the feed drive at the start of the path. This validates the theory developed for the "generation of the tool motion along the tool path" and the "approach procedure".

- The tool motion along the tool path. This validates the theory developed for the "generation of the tool motion along the tool path" and the "regular procedure".
- The deceleration of the feed drive at the end of the path. This validates the theory developed for the "generation of the feed drive deceleration at the end of the path". For the long tool path, the feed drive has enough time to accelerate at the start of the path and reach the maximum allowable feed rate as predicted (figures 6.17c and 6.17d). For the short tool path, the feed drive does not have enough time to reach the maximum allowable feed rate resulting in a "triangular profile" of the velocity (figures 6.16c and 6.16d).
- The simulation generates an actual feed rate that is not allowed to reach the commanded feed rate as shown in figure 6.18.

Considering the cutting force, a force spike is observed when the tool enters the workpiece in all the experiments. This spike is generated by the impact between the tool and the workpiece when initial contact is made. It is considered a transient phenomenon and is not modeled by the mechanistic force model used in this study.

The magnitude of the cutting force drops below the target force during certain regions within the workpiece. Within these regions, the geometry of the workpiece changes rapidly. The feed drive system limitations do not allow the feed rate to change fast enough to maintain a constant cutting force.

In all other regions of the workpiece, there is good agreement between the predicted and experimental cutting force. The results of the optimized feed scheduling strategy offer a relatively uniform tool load without high force peaks. This results in stable cutting conditions and satisfactory process integrity. Good regulation of the cutting force is achieved. The actual feed rate of the feed drive does not overshoot the target feed rate. Therefore the experimental cutting forces are less than the predicted cutting force. This guarantees that the maximum load on the tool will not be exceeded, and prevents tool breakage.

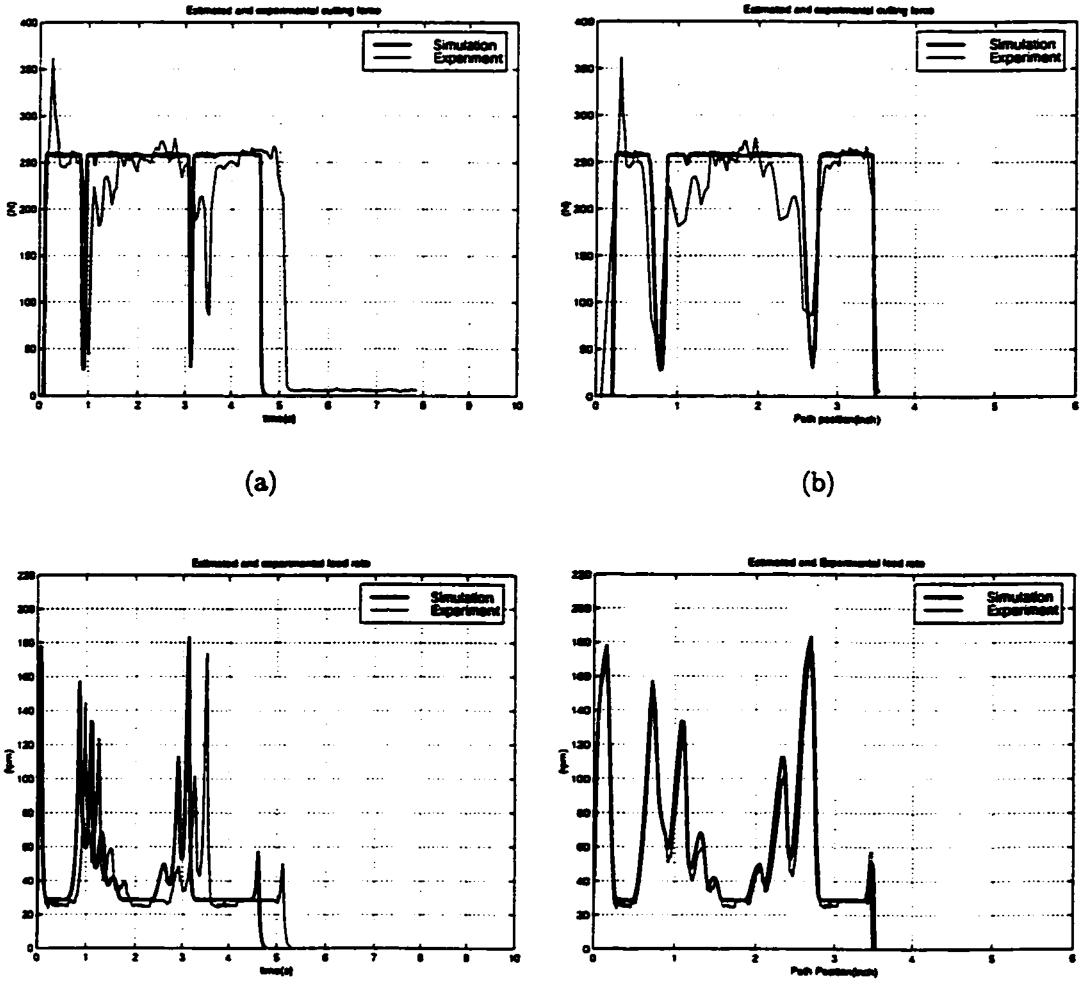
The estimated and experimental machining time, for the short tool path length, are equal to 4.80s and 5.29s, respectively. For the long tool path length, the machining times are equal to 5.41s and 6.00s, respectively. Applying the optimized feed scheduling strategy improves the productivity for these tests by 31% and 48%, respectively.

These results validate the effectiveness of the theory developed for the optimized feed scheduling strategy.

6.3.5 Summary of Feed Scheduling Strategies

In this section, the different feed scheduling strategies are compared. The criteria for comparison are the production saving, the machining integrity, and the machining safety.

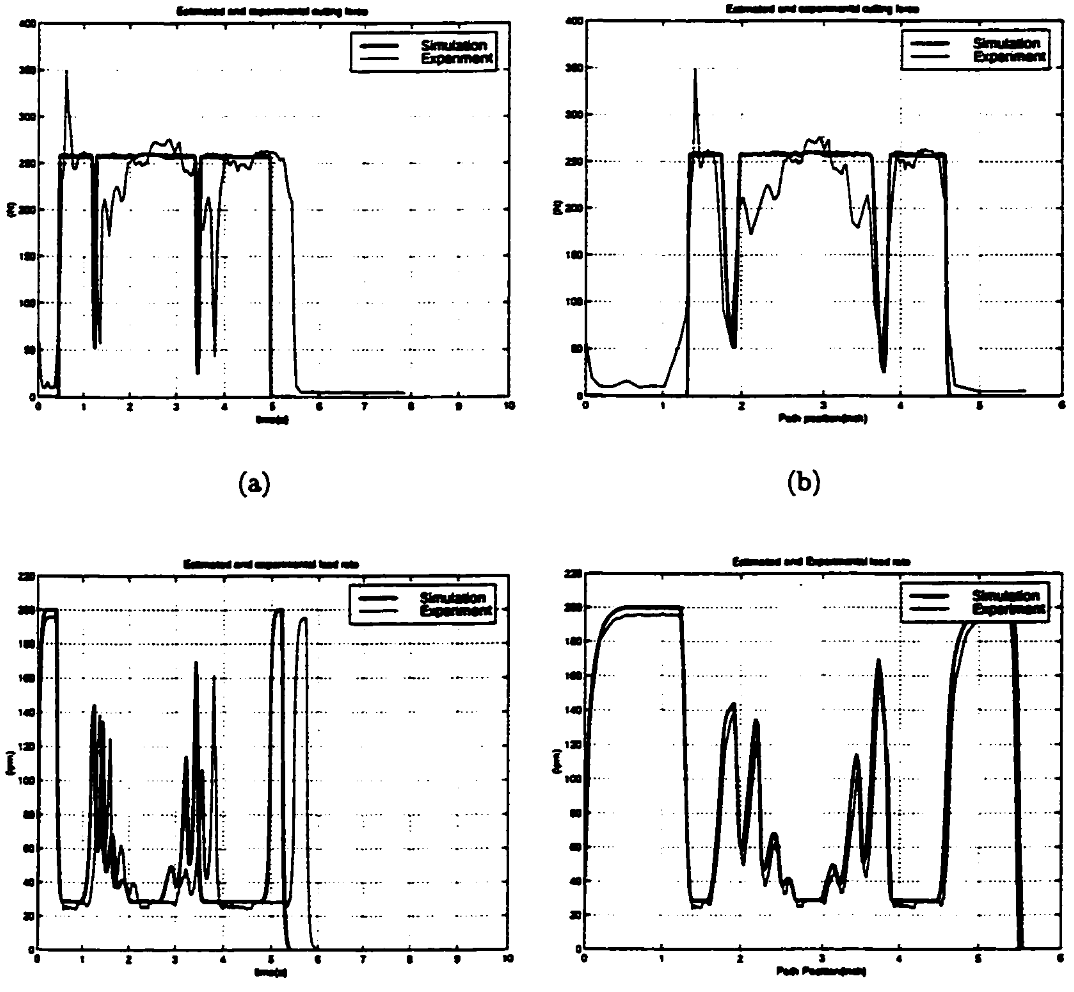
Production saving The two sets of experiments, with short and long tool paths, were introduced to demonstrate the production saving when the air cutting-to-cutting ratio is high. This productivity gain is effective for the different feed scheduling strategy (figure 6.19). The feed scheduling strategy based on control points (FSCP) achieves the best production enhancement. This is followed by the optimized feed scheduling strategy (OFSS) and finally by the feed scheduling strategy based on the Acc/Dec time constant (FSAD). In all cases a productivity



(a) (b)

(c) (d)

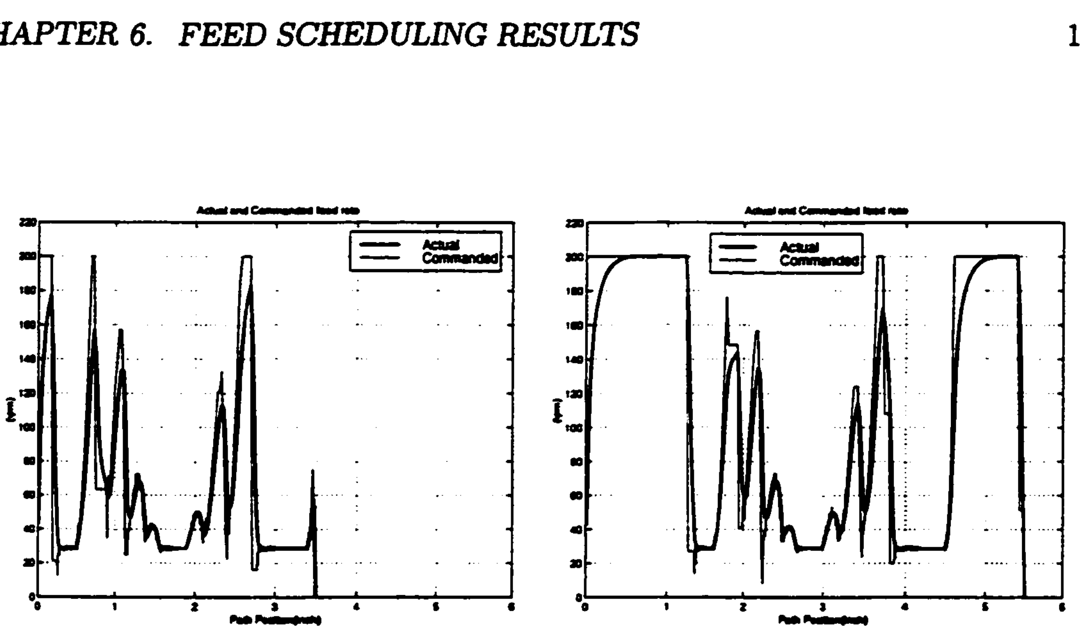
Figure 6.16: Optimized feed scheduling strategy, short tool path, (a) Cutting force: Time representation, (b) Cutting force: Spatial representation, (c) Feed rate: Time representation, (d) Feed rate: Spatial representation



(a) (b)

(c) (d)

Figure 6.17: Optimized feed scheduling strategy, long tool path, (a) Cutting force: Time representation, (b) Cutting force: Spatial representation, (c) Feed rate: Time representation, (d) Feed rate: Spatial representation



(a) (b)

Figure 6.18: Commanded and actual feed rate generated by the simulation, (a) Short tool path, (b) Long tool path

gain was achieved by applying a feed scheduling strategy.

Machining Integrity Machining integrity is achieved when the cutting force can be accurately predicted and well regulated. The FSCP did not predict the cutting force well or provide good regulation of the force. This is explained by the poor prediction of the actual feed rate. The FSAD provides a good prediction of the force. However, it does not provide good regulation of the cutting force. This is due to the constraint that the actual feed rate must reach the commanded feed rate. This constraint is very conservative because the feed drive system is not pushed to its limits to make the feed rate changes as fast as the variations of the cutting geometry. The OFSS is the only feed scheduling strategy that achieves good cutting force prediction and provides good regulation of the cutting force.

Machining Safety Machining safety is achieved when the cutting force is predicted

	Machining Integrity/Safety			Machining Time				Production Saving	
	Pred.		Reg.	Short path		Long path		Short path	Long path
	force	Feed	force	Est.	Exp.	Est.	Exp.		
No feed scheduling	Good	Good	-	7.25	7.67	11.46	11.54	-	-
Feed scheduling FSCP	Poor	Poor	Poor	4.50	4.99	5.10	5.67	35 %	51 %
Feed scheduling FSAD	Good	Good	Poor	5.66	5.82	6.38	6.65	24 %	42 %
Feed scheduling OFSS	Good	Good	Good	4.80	5.29	5.41	6.00	31 %	48 %

Table 6.2: Summary of feed scheduling strategies

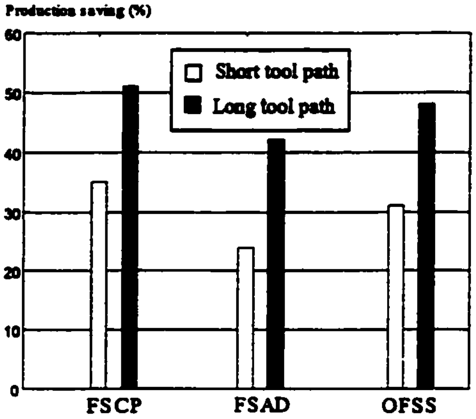


Figure 6.19: Production Saving

accurately. Only the FSCP does not satisfy this criterion. The machining safety is guaranteed when the machining integrity is achieved. However, when good force regulation is not achieved (i.e., the FSAD or the cutting experiment with nominal feed rate), the machining operation can still be safe.

These criteria have been summarized in table 6.2. The optimized feed scheduling strategy is the unique strategy that improves simultaneously the production saving while guaranteeing the machining integrity.

6.4 Summary

This chapter demonstrates the feed scheduling concept for the methodologies presented in chapter 4. A case study for process optimization of machining an airfoil-like surface was used for demonstration. Based on the predicted instantaneous chip load and/or a specified force constraint, feed rate scheduling is utilized to increase metal removal rate. The feed rate scheduling implementation results in a 30% reduction in machining time for the airfoil-like surface without any sacrifice in the surface quality or part geometry.

This chapter also demonstrated the methodologies for incorporating the feed drive performance capabilites with feed scheduling. A comparative study was performed for feed scheduling based on control points, Acc/Dec time constant, and optimized strategy. Both simulated and experimental results were given for each feed scheduling strategy. The criteria for comparison were the production savings, the machining integrity, and the machining safety. The optimized feed scheduling strategy was the unique strategy that improves simultaneously the production saving while guaranteeing the machining integrity.

Chapter 7

Conclusions and Future Research

7.1 Research Summary

Complex scuptured surfaces are common features in parts and/or components that are produced in the die and mold, automotive, aerospace, and appliances industries. A generic simulation approach for multi-axis machining of complex sculptured surface parts has been developed in this thesis. The simulation approach includes a geometric modeling scheme that is generic for multi-axis machining. The approach also includes a comprehensive force modeling methodology for multi-axis machining.

The geometric modeling scheme was developed to provide:

- A generalized approach for representing an arbitrary cutting edge design.
- A generalized approach for representing the local surface topology of the part, without using a computationally expensive CAD solid modeler.
- Implementation of a simple methodology for determining the tool/part interaction for complex sculptured surface machining.

The geometric modeling scheme permits an arbitrary tool path (CLData file) to be specified. The geometric modeling scheme also allows multiple machining operations to be integrated within a single simulation (ie. specified by multiple tools and CLData files). This allows the desired machining operation (roughing, semi-finishing, finishing) to be simulated without having to simulate previous machining operations. For example: to simulate a finishing operation, neither the semi-finish or roughing operation need to be simulated first. The local surface topology of the part is constructed from only the tool path specifications given by the semi-finish and roughing CLData files. This methodology provides an accurate representation of the local surface topology and substantially reduces the computation time for the simulation. The solid modeling approaches that have been presented in the literature have to update the part geometry after each tool movement in the CLData file. The solid modeling approaches can only simulate a single machining operation. To simulate multiple machining operations, each machining operation must be simulated individually. For the example cited above, both the roughing and semi-finishing operations would have to be simulated to provide the updated part geometry for the finishing operation simulation. The computational work load is substantially increased when using a solid modeling approach.

A comprehensive cutting force model for multi-axis machining was presented. The cutting force model considers both normal and friction components of force on the rake and clearance faces on the cutting edge. A systematic model calibration procedure was developed to incorporate the effects of varying cutting edge geometry, cutting speeds, and feed rates. A series of model calibration tests were performed. Model verification was conducted with the cutting force calibration coefficients determined from the model calibration tests. The verification results demonstrate that the predicted forces are within 5% of experimentally measured forces.

A continuous dynamic simulation methodology for multi-axis machining was developed. The methodology was formulated to include:

- A minimum simulation time period.
- The instantaneous location of the cutting edge based on the original tool centre (defined by the current tool path location) and the current deflection of the tool.
- The instantaneous chip thickness determined from the feed/tooth and the deflection of the previous N tooth periods (regenerative model).
- The deflection effects of previous tool path segments.

The continuous dynamic process simulation methodology was compared with both static and dynamic simulation methodologies based on control points. The comparison was for linear cuts with constant and variable tool/workpiece immersion. This comparison showed that the modeling methodologies based on control points only offer a piece-wise continous representation of the static and dynamic cutting force and tool deflection over a given tool path. The accuracy of the cutting force prediction and the calculation of tool deflection using methodologies based on control points degrades for variable tool/workpiece immersion. The continous dynamic simulation methodology demonstrates a good representation of the static and dynamic cutting force and tool deflection for variable tool/workpiece immersion. The continous dynamic process simulation methodology was also demonstrated for a complex sculptured surface machining operation. The rough machining operation for an airfoil-like surface was presented. The simulation results demonstrate how the continuous dynamic process simulation methodology is capable of predicting the cutting force and tool deflection for variable tool/workpiece immersions that occur during complex sculptured surface machining operations

The generic simulation approach for multi-axis machining has been demonstrated as a process optimization tool. Feed scheduling was used to demonstrate the process optimization for multi-axis machining. A feed scheduling methodology for multi-axis machining was developed. The feed scheduling methodology was formulated based on maximum chipload and maximum force constraints. A case study for process optimization of machining an airfoil-like surface was used for demonstration. Based on the predicted instantaneous chip load and/or a specified force constraint, feed rate scheduling was utilized to increase metal removal rate. The feed rate scheduling implementation results in a 30% reduction in machining time for the airfoil-like surface without any sacrifice in the surface quality or part geometry.

The machine tool feed drive performance capabilities were integrated with the feed scheduling methodology. Two strategies were developed. The first strategy guaranteed that the scheduled feed rates were attainable within the machine Acc/Dec time constant, τ_{total}. The second strategy optimized the use of the feed drive capabilities while tracking the changes in cutting geometry along the tool path. A comparative study was performed for feed scheduling based on control points, Acc/Dec time constant, and optimized strategy. Both simulated and experimental results were given for each feed scheduling strategy. The criteria for comparison was the production savings, the machining integrity, and the machining safety. The optimized feed scheduling strategy was the unique strategy that improves simultaneously the production saving while guaranteeing the machining integrity.

The generic simulation approach for multi-axis machining was integrated with Internet technologies to provide an Internet enabled process simulation and optimization environment. The research contributions related to the Internet implementation of the multi-axis machining simulation include:

- The Internet provides an ideal medium to provide wide spread access to advanced process simulation software. Access to machining process simulation software has typically been limited to individual research institutions. The Internet based implementation provided here allows the simulation software to be accessed by other research institutions and by industrial users around the world.
- The interactive nature of the Internet based implementation of the multi-axis machining simulation offers the user valuable insight into the machining process. This information can be used for reducing or eliminating the need for expensive process trials. Specific process problems can also be diagnosed more efficiently when this information is available. Having this information allows the user to optimize the machining process by testing different process parameters (feed, speed, tool paths, etc.).

7.2 Future Research Directions

The generic simulation approach for multi-axis machining has revealed certain key areas that need further investigation. There are both fundamental research issues and practical implementation mechanisms that should be addressed in future efforts.

The mechanistic force model presented in chapter 3 has been formulated assuming a sharp tool. Tool wear has not been considered in the model formulation. During practical machining operations the tool wears as a natural consequence of the tool/workpiece interaction. As the amount of tool wear increases, the accuracy of the cutting force predicted by the mechanistic force model will degrade. The effects of tool wear should be incorporated within the modeling methodology. There are different mechanisms of tool wear that occur on both the rake and clearance faces of the cutting edge. Cutting process temperatures also affect tool wear. Tool wear and its effects on model performance will need to be further investigated. As an alternative solution for incorporating the effects of tool wear, the model calibration procedure may be applied for tools with varying tool wear. The value of the derived calibration coefficient will include the effects of tool wear. Currently, the calibration coefficient is a function of cutting edge geometry, cutting speed, and feed rate. The functional form of the calibration coefficients may need to also include tool wear. The effects of tool wear on the model calibration coefficients should also be investigated.

The model calibration procedure determines the coefficient values based on a regression analysis. Each calibration coefficient is a function of three parameters; cutting edge geometry, cutting speed, and feed rate. Although, the effect on the calibration coefficient for each of these parameters has been demonstrated, the relative importance of each parameter was not investigated. As a further research issue, a parameter sensitivity analysis should be carried out for each of these parameters. The calibration coefficients are valid for a specific tool/workpiece material combination. The basic material properties (yield stress, etc.) for the tool and workpiece are not incorporated within the calibration coefficient function. Incorporating material properties should be investigated.

The process constraints used for feed rate scheduling presented in chapter 4 include maximum chipload and maximum force constraints. Both chipload and cutting force are often intermediate process variables but not the ultimate process variables carrying functional importance for the end user. Other process constraints such as finish, form error, cycle time, or part integrity provide a more directly usable set of process constraints for feed rate scheduling. Each of the additional process constraints listed above can be expressed as functions of either the chipload or force constraint. However, the feed rate scheduling methodology should be reformulated to include these additional process constraints.

The feed rate scheduling strategies that incorporated the machine tool feed drive performance limitations have only been developed and verified for a single axis motion. The principles demonstrated in this work should be extended to mult-axis contour machining operations. The identification procedure given in Appendix D can be used to determine the Acc/Dec profile for each machine tool feed drive. Contour machining requires the coordinated motion of multiple axis. The coordinated motion is scheduled by the machine tool interpolator. There may be some cross-coupling effects that may need to be identified and incorporated with the feed rate scheduling strategy. An identification procedure similar to the method given in Appendix D may need to be investigated.

There are specific algorithms that can be optimized to reduce simulation times. The algorithms used to implement the modeling methodologies were developed with the philosophy, 'Make it work first, then optimize'. The algorithms that required the largest computational times were related to the intersection calculation between the local surface topology and the cutting edge. Further investigation is required to optimize these algorithms.

The local surface topology of the part is generated from the initial part surface topology and previous machining operations. Currently, the initial part geometry is given as a rectangular block. This initial geometry of the part is valid for many machining operations. However, the initial part geometry may be generated by other manufacturing processes. In this case the key is to present the simulation with an adequate representation of the initial part surface topology. This information can be provided by a CAD model of the part or from CMM data. The surface representation could be presented to the simulation as either primitives (planes, conics, etc.) or complex surface representations (bezier, NURBS, etc.).

There is a lack of available software tools for manufacturing process development. The Internet-based implementation of the multi-axis machining process simulation and optimization can be combined with other related machining technologies to provide a formalized system for designing and proofing manufacturing processes. Process develoment can occur iteratively, without the need for lengthy and/or expensive process trials. Implementing these development tools using Internet technologies has several advantages. It allows all of the development tools to be in a centralized location, allowing for quick and reliable upgrade paths. Internet accessibility also gives quick and easy access to these development tools to many manufacturing engineers around the world.

Appendix A

Force Model Reformulation

$$\begin{aligned}
\vec{\Im}_p(\theta) &= \vec{\Im}_{nr}(\theta) + \vec{\Im}_{fr}(\theta) + \vec{\Im}_{ne}(\theta) + \vec{\Im}_{fe}(\theta) \\
&= \int_{z1}^{z2} \{K_{nr} \cdot [-\vec{n}_r(\theta, z) + K_{fr} \cdot \vec{n}_{fr}(\eta_c, \theta, z)] \cdot \delta A_r(\theta, z) \qquad \text{(A.1)} \\
&\quad +K_{ne} \cdot [-\vec{n}_e(\theta, z) + K_{fe} \cdot \vec{n}_{fe}(\theta, z)] \cdot \delta S
\end{aligned}$$

if $\vec{n}_{fr}(\eta_c, \theta, z) = \vec{b}(\theta, z) \cdot \cos\eta_c + \vec{r}(\theta, z) \cdot \sin\eta_c$

$$\begin{aligned}
\vec{\Im}_p(\theta) &= \int_{z1}^{z2} \{K_{nr} \cdot [-\vec{n}_r(\theta, z) + K_{fr} \cdot (\vec{b}(\theta, z) \cdot \cos\eta_c + \vec{r}(\theta, z) \cdot \sin\eta_c)] \cdot \delta A_r(\theta, z) \\
&\quad +K_{ne} \cdot [-\vec{n}_e(\theta, z) + K_{fe} \cdot \vec{n}_{fe}(\theta, z)] \cdot \delta S \qquad \text{(A.2)}
\end{aligned}$$

Rearranging using:

$$K_r = \begin{bmatrix} K_{nr} \\ K_{nr} \cdot K_{fr} \cdot \cos \eta_c \\ K_{nr} \cdot K_{fr} \cdot \sin \eta_c \end{bmatrix} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix} \tag{A.3}$$

$$K_f = \begin{bmatrix} K_{ne} \\ K_{ne} \cdot K_{fe} \end{bmatrix} = \begin{bmatrix} K_4 \\ K_5 \end{bmatrix} \tag{A.4}$$

$$\hat{R}(\theta, z) = \begin{bmatrix} -n_{rx} & b_x & \tau_x \\ -n_{ry} & b_y & \tau_y \\ -n_{rz} & b_z & \tau_z \end{bmatrix} \tag{A.5}$$

$$\hat{E}(\theta, z) = \begin{bmatrix} -n_{ex} & n_{fex} \\ -n_{ey} & n_{fey} \\ -n_{ez} & n_{fez} \end{bmatrix} \tag{A.6}$$

$$\vec{\Im}_p(\theta) \;=\; \int_{z1}^{z2} \{\hat{R}(\theta, z) \cdot K_r \cdot \delta A_r(\theta, z) + \hat{E}(\theta, z) \cdot K_f \cdot \delta S \tag{A.7}$$

For a given angular position of cutter, θ, the coordinate forces ($\Im_{px}$,$\Im_{py}$,$\Im_{pz}$) are given by:

$$\begin{aligned} \Im_{px} &= \sum_{z1}^{z2}\{[K_1 \cdot -n_{rx}(z) + K_2 \cdot b_x(z) + K_3 \cdot \tau_x(z)] \cdot \delta A_r(z) \\ &\quad +[K_4 \cdot -n_{ex}(z) + K_5 \cdot n_{fex}(z)] \cdot \delta S(z) \\ &= \sum_{z1}^{z2}\{K_1 \cdot a_1(z)(z) + K_2 \cdot a_2(z)(z) + K_3 \cdot a_3(z)(z) \\ &\quad +K_4 \cdot a_4(z)(z) + K_5 \cdot a_5(z)(z)\} \end{aligned} \tag{A.8}$$

where

$$\begin{aligned} a_1(z)(z) &= -n_{rx}(z) \cdot \delta A_r(z) \\ a_2(z)(z) &= b_x(z) \cdot \delta A_r(z) \\ a_3(z)(z) &= \tau_x(z) \cdot \delta A_r(z) \\ a_4(z)(z) &= -n_{ex}(z) \cdot \delta A_r(z) \\ a_5(z)(z) &= n_{fex}(z) \cdot \delta A_r(z) \end{aligned}$$

$$\begin{aligned} \Im_{py} &= \sum_{z1}^{z2}\{[K_1 \cdot -n_{ry}(z) + K_2 \cdot b_y(z) + K_3 \cdot \tau_y(z)] \cdot \delta A_r(z) \\ &\quad +[K_4 \cdot -n_{ey}(z) + K_5 \cdot n_{fey}(z)] \cdot \delta S(z) \\ &= \sum_{z1}^{z2}\{K_1 \cdot b_1(z)(z) + K_2 \cdot b_2(z)(z) + K_3 \cdot b_3(z)(z) \\ &\quad +K_4 \cdot b_4(z)(z) + K_5 \cdot b_5(z)(z)\} \end{aligned} \tag{A.9}$$

where

$$
\begin{aligned}
b_1(z)(z) &= -n_{\tau y}(z) \cdot \delta A_r(z) \\
b_2(z)(z) &= b_y(z) \cdot \delta A_r(z) \\
b_3(z)(z) &= \tau_y(z) \cdot \delta A_r(z) \\
b_4(z)(z) &= -n_{ey}(z) \cdot \delta A_r(z) \\
b_5(z)(z) &= n_{fey}(z) \cdot \delta A_r(z)
\end{aligned}
$$

$$
\begin{aligned}
\Im_{pz} &= \sum_{z1}^{z2} \{[K_1 \cdot -n_{\tau z}(z) + K_2 \cdot b_z(z) + K_3 \cdot \tau_z(z)] \cdot \delta A_r(z) \\
&\quad +[K_4 \cdot -n_{ez}(z) + K_5 \cdot n_{fez}(z)] \cdot \delta S(z) \qquad \text{(A.10)} \\
&= \sum_{z1}^{z2} \{K_1 \cdot c_1(z)(z) + K_2 \cdot c_2(z)(z) + K_3 \cdot c_3(z)(z) \\
&\quad +K_4 \cdot c_4(z)(z) + K_5 \cdot c_5(z)(z)\}
\end{aligned}
$$

where

$$
\begin{aligned}
c_1(z)(z) &= -n_{\tau z}(z) \cdot \delta A_r(z) \\
c_2(z)(z) &= b_z(z) \cdot \delta A_r(z) \\
c_3(z)(z) &= \tau_z(z) \cdot \delta A_r(z) \\
c_4(z)(z) &= -n_{ez}(z) \cdot \delta A_r(z) \\
c_5(z)(z) &= n_{fez}(z) \cdot \delta A_r(z)
\end{aligned}
$$

In all of the above equations, the only unknown variables are $\{K_1, K_2, K_3, K_4, K_5\}$.

These are constant across the summations, therefore from the properties of summations:

$$\begin{aligned}
\Im_{px} &= K_1 \sum_{z1}^{z2} a_1(z)(z) + K_2 \sum_{z1}^{z2} a_2(z)(z) + K_3 \sum_{z1}^{z2} a_3(z)(z) \\
&\quad + K_4 \sum_{z1}^{z2} a_4(z)(z) + K_5 \sum_{z1}^{z2} a_5(z)(z) \\
\Im_{py} &= K_1 \sum_{z1}^{z2} b_1(z)(z) + K_2 \sum_{z1}^{z2} b_2(z)(z) + K_3 \sum_{z1}^{z2} b_3(z)(z) \\
&\quad + K_4 \sum_{z1}^{z2} b_4(z)(z) + K_5 \sum_{z1}^{z2} b_5(z)(z) \qquad \text{(A.11)} \\
\Im_{pz} &= K_1 \sum_{z1}^{z2} c_1(z)(z) + K_2 \sum_{z1}^{z2} c_2(z)(z) + K_3 \sum_{z1}^{z2} c_3(z)(z) \\
&\quad + K_4 \sum_{z1}^{z2} c_4(z)(z) + K_5 \sum_{z1}^{z2} c_5(z)(z)
\end{aligned}$$

Note that the $a_i(z)$, $b_i(z)$, $c_i(z)$ are strictly geometric quantities and can be precalculated as:

$$\begin{aligned}
A_i &= \sum_{z1}^{z2} a_i(z) \\
B_i &= \sum_{z1}^{z2} b_i(z) \qquad \text{(A.12)} \\
C_i &= \sum_{z1}^{z2} c_i(z)
\end{aligned}$$

(A.13)

Giving

$$\begin{aligned}
\Im_{px} &= K_1 \cdot A_1 + K_2 \cdot A_2 + K_3 \cdot A_3 + K_4 \cdot A_4 + K_5 \cdot A_5 \\
\Im_{py} &= K_1 \cdot B_1 + K_2 \cdot B_2 + K_3 \cdot B_3 + K_4 \cdot B_4 + K_5 \cdot B_5 \qquad \text{(A.14)} \\
\Im_{pz} &= K_1 \cdot C_1 + K_2 \cdot C_2 + K_3 \cdot C_3 + K_4 \cdot C_4 + K_5 \cdot C_5
\end{aligned}$$

Appendix B

Calibration Procedure I

This appendix gives the mathematical details for performing the multiple parameter linear regression analysis (LSE) to determine the coefficients $\{K_1, K_2, K_3, K_4, K_5\}$.

In general matrix form:

$$Y = X\beta + E \tag{B.1}$$

where Y is the vector of responses, β is the vector of model parameters, E is the vector of random errors, and X is the model specification matrix. The errors, E are assumed to be uncorrelated having zero means and the same variance σ^2.

The normal equation of the general linear model is:

$$(X' \cdot X) \cdot \beta = X' \cdot Y \tag{B.2}$$

The solution to the normal equation is:

$$\beta = (X' \cdot X)^{-1} \cdot X' \cdot Y \tag{B.3}$$

The sum of squares at the minimum is

$$S(\beta) = Y'Y - \beta' X' X \beta \tag{B.4}$$

An estimation of residual variance [14], σ^2, is given as

$$s^2 = \frac{S(\beta)}{n-k} \tag{B.5}$$

where n is the number of observations and k is the number of model parameters.

A $1-\epsilon$ confidence region for β [14] can be defined as

$$S = S(\beta)\left[1 + \frac{k}{n-k} F_\alpha(k, n-k)\right] \tag{B.6}$$

where $F_\alpha(k, n-k)$ is the significance point of the F distribution with k and $n-k$ degrees of freedom.

For the current case the model takes the form:

$$\begin{aligned} \Im_{mx} &= K \cdot A + E_x \\ \Im_{my} &= K \cdot B + E_y \\ \Im_{mz} &= K \cdot C + E_z \end{aligned} \tag{B.7}$$

The Sum Squared Error (SSE) is taken as:

$$
\begin{aligned}
SSE &= \sum_i^n e_i^2 \\
&= \sum_i^n (e_{xi}^2 + e_{yi}^2 + ez) \\
&= \sum_i^n e_{xi}^2 + \sum_i^n e_{yi}^2 + \sum_i^n e_{zi}^2 \qquad\qquad (B.8) \\
&= \sum_i^n [\Im_{mxi} - (K_1 \cdot A_{1i} + K_2 \cdot A_{2i} + K_3 \cdot A_{3i} + K_4 \cdot A_{4i} + K_5 \cdot A_{5i})]^2 \\
&= \sum_i^n [\Im_{myi} - (K_1 \cdot B_{1i} + K_2 \cdot B_{2i} + K_3 \cdot B_{3i} + K_4 \cdot B_{4i} + K_5 \cdot B_{5i})]^2 \\
&= \sum_i^n [\Im_{mzi} - (K_1 \cdot C_{1i} + K_2 \cdot C_{2i} + K_3 \cdot C_{3i} + K_4 \cdot C_{4i} + K_5 \cdot C_{5i})]^2
\end{aligned}
$$

The Sum Square Error is minimized when

$$
\frac{\delta SSE}{\delta K_i} = 0, \ (i = 1, \cdots, 5)
$$

The partial derivatives are given as:

$$
\begin{aligned}
\frac{\delta SSE}{\delta K_1} &= -2 \cdot \sum_i^n [\Im_{mxi} - (K_1 \cdot A_{1i} + K_2 \cdot A_{2i} + K_3 \cdot A_{3i} + K_4 \cdot A_{4i} + K_5 \cdot A_{5i})] \cdot A_{1i} \\
&= -2 \cdot \sum_i^n [\Im_{myi} - (K_1 \cdot B_{1i} + K_2 \cdot B_{2i} + K_3 \cdot B_{3i} + K_4 \cdot B_{4i} + K_5 \cdot B_{5i})] \cdot B_{1i} \\
&= -2 \cdot \sum_i^n [\Im_{mzi} - (K_1 \cdot C_{1i} + K_2 \cdot C_{2i} + K_3 \cdot C_{3i} + K_4 \cdot C_{4i} + K_5 \cdot C_{5i})] \cdot C_{1i}
\end{aligned}
\tag{B.9}
$$

$$
\begin{aligned}
\frac{\delta SSE}{\delta K_2} &= -2 \cdot \sum_i^n [\Im_{mxi} - (K_1 \cdot A_{1i} + K_2 \cdot A_{2i} + K_3 \cdot A_{3i} + K_4 \cdot A_{4i} + K_5 \cdot A_{5i})] \cdot A_{2i} \\
&= -2 \cdot \sum_i^n [\Im_{myi} - (K_1 \cdot B_{1i} + K_2 \cdot B_{2i} + K_3 \cdot B_{3i} + K_4 \cdot B_{4i} + K_5 \cdot B_{5i})] \cdot B_{2i} \\
&= -2 \cdot \sum_i^n [\Im_{mzi} - (K_1 \cdot C_{1i} + K_2 \cdot C_{2i} + K_3 \cdot C_{3i} + K_4 \cdot C_{4i} + K_5 \cdot C_{5i})] \cdot C_{2i}
\end{aligned}
\tag{B.10}
$$

$$
\begin{aligned}
\frac{\delta SSE}{\delta K_3} &= -2 \cdot \sum_i^n [\Im_{mxi} - (K_1 \cdot A_{1i} + K_2 \cdot A_{2i} + K_3 \cdot A_{3i} + K_4 \cdot A_{4i} + K_5 \cdot A_{5i})] \cdot A_{3i} \\
&= -2 \cdot \sum_i^n [\Im_{myi} - (K_1 \cdot B_{1i} + K_2 \cdot B_{2i} + K_3 \cdot B_{3i} + K_4 \cdot B_{4i} + K_5 \cdot B_{5i})] \cdot B_{3i} \\
&= -2 \cdot \sum_i^n [\Im_{mzi} - (K_1 \cdot C_{1i} + K_2 \cdot C_{2i} + K_3 \cdot C_{3i} + K_4 \cdot C_{4i} + K_5 \cdot C_{5i})] \cdot C_{3i}
\end{aligned}
\tag{B.11}
$$

$$\begin{aligned}\frac{\delta SSE}{\delta K_4} &= -2\cdot\sum_i^n[\Im_{mxi}-(K_1\cdot A_{1i}+K_2\cdot A_{2i}+K_3\cdot A_{3i}+K_4\cdot A_{4i}+K_5\cdot A_{5i})]\cdot A_{4i}\\ &= -2\cdot\sum_i^n[\Im_{myi}-(K_1\cdot B_{1i}+K_2\cdot B_{2i}+K_3\cdot B_{3i}+K_4\cdot B_{4i}+K_5\cdot B_{5i})]\cdot B_{4i}\\ &= -2\cdot\sum_i^n[\Im_{mzi}-(K_1\cdot C_{1i}+K_2\cdot C_{2i}+K_3\cdot C_{3i}+K_4\cdot C_{4i}+K_5\cdot C_{5i})]\cdot C_{4i}\end{aligned} \tag{B.12}$$

$$\begin{aligned}\frac{\delta SSE}{\delta K_5} &= -2\cdot\sum_i^n[\Im_{mxi}-(K_1\cdot A_{1i}+K_2\cdot A_{2i}+K_3\cdot A_{3i}+K_4\cdot A_{4i}+K_5\cdot A_{5i})]\cdot A_{5i}\\ &= -2\cdot\sum_i^n[\Im_{myi}-(K_1\cdot B_{1i}+K_2\cdot B_{2i}+K_3\cdot B_{3i}+K_4\cdot B_{4i}+K_5\cdot B_{5i})]\cdot B_{5i}\\ &= -2\cdot\sum_i^n[\Im_{mzi}-(K_1\cdot C_{1i}+K_2\cdot C_{2i}+K_3\cdot C_{3i}+K_4\cdot C_{4i}+K_5\cdot C_{5i})]\cdot C_{5i}\end{aligned} \tag{B.13}$$

Recall that the Sum Square Error is minimized when

$$\frac{\delta SSE}{\delta K_i} = 0,\ (i=1,\cdots,5)$$

gives the following system of equations

1)

$\sum_i^n(A_{1i}\cdot\Im_{mxi}+B_{1i}\cdot\Im_{myi}+C_{1i}\cdot\Im_{mzi}) =$

$K_1\cdot[\sum_i^n(A_{1i}^2+B_{1i}^2+C_{1i}^2)]$

$+K_2\cdot[\sum_i^n(A_{1i}\cdot A_{2i}+B_{1i}\cdot B_{2i}+C_{1i}\cdot C_{2i})]$

$+K_3\cdot[\sum_i^n(A_{1i}\cdot A_{3i}+B_{1i}\cdot B_{3i}+C_{1i}\cdot C_{3i})]$

$+K_4\cdot[\sum_i^n(A_{1i}\cdot A_{4i}+B_{1i}\cdot B_{4i}+C_{1i}\cdot C_{4i})]$

$+K_5\cdot[\sum_i^n(A_{1i}\cdot A_{5i}+B_{1i}\cdot B_{5i}+C_{1i}\cdot C_{5i})]$

Simplifying,

$$Y_1 = K_1 \cdot X_{1,1} + K_2 \cdot X_{1,2} + K_3 \cdot X_{1,3} + K_4 \cdot X_{1,4} + K_5 \cdot X_{1,5} \qquad \text{(B.14)}$$

where

$$\begin{aligned}
Y_1 &= \textstyle\sum_i^n (A_{1i} \cdot \Im_{mxi} + B_{1i} \cdot \Im_{myi} + C_{1i} \cdot \Im_{mzi}) \\
X_{1,1} &= \textstyle\sum_i^n (A_{1i}^2 + B_{1i}^2 + C_{1i}^2) \\
X_{1,2} &= \textstyle\sum_i^n (A_{1i} \cdot A_{2i} + B_{1i} \cdot B_{2i} + C_{1i} \cdot C_{2i}) \\
X_{1,3} &= \textstyle\sum_i^n (A_{1i} \cdot A_{3i} + B_{1i} \cdot B_{3i} + C_{1i} \cdot C_{3i}) \\
X_{1,4} &= \textstyle\sum_i^n (A_{1i} \cdot A_{4i} + B_{1i} \cdot B_{4i} + C_{1i} \cdot C_{4i}) \\
X_{1,5} &= \textstyle\sum_i^n (A_{1i} \cdot A_{5i} + B_{1i} \cdot B_{5i} + C_{1i} \cdot C_{5i})
\end{aligned}$$

2)

$$\sum_i^n (A_{2i} \cdot \Im_{mxi} + B_{2i} \cdot \Im_{myi} + C_{2i} \cdot \Im_{mzi}) =$$

$$+K_1 \cdot [\sum_i^n (A_{2i} \cdot A_{1i} + B_{2i} \cdot B_{1i} + C_{2i} \cdot C_{1i})]$$

$$+K_2 \cdot [\sum_i^n (A_{2i}^2 + B_{2i}^2 + C_{2i}^2)]$$

$$+K_3 \cdot [\sum_i^n (A_{2i} \cdot A_{3i} + B_{2i} \cdot B_{3i} + C_{2i} \cdot C_{3i})]$$

$$+K_4 \cdot [\sum_i^n (A_{2i} \cdot A_{4i} + B_{2i} \cdot B_{4i} + C_{2i} \cdot C_{4i})]$$

$$+K_5 \cdot [\sum_i^n (A_{2i} \cdot A_{5i} + B_{2i} \cdot B_{5i} + C_{2i} \cdot C_{5i})]$$

Simplifying.

$$Y_2 = K_1 \cdot X_{2,1} + K_2 \cdot X_{2,2} + K_3 \cdot X_{2,3} + K_4 \cdot X_{2,4} + K_5 \cdot X_{2,5} \qquad \text{(B.15)}$$

where

$$Y_2 = \sum_i^n (A_{2i} \cdot \Im_{mxi} + B_{2i} \cdot \Im_{myi} + C_{2i} \cdot \Im_{mzi})$$

$$X_{2,1} = \sum_i^n (A_{2i} \cdot A_{1i} + B_{2i} \cdot B_{1i} + C_{2i} \cdot C_{1i})$$

$$X_{2,2} = \sum_i^n (A_{2i}^2 + B_{2i}^2 + C_{2i}^2)$$

$$X_{2,3} = \sum_i^n (A_{2i} \cdot A_{3i} + B_{2i} \cdot B_{3i} + C_{2i} \cdot C_{3i})$$

$$X_{2,4} = \sum_i^n (A_{2i} \cdot A_{4i} + B_{2i} \cdot B_{4i} + C_{2i} \cdot C_{4i})$$

$$X_{2,5} = \sum_i^n (A_{2i} \cdot A_{5i} + B_{2i} \cdot B_{5i} + C_{2i} \cdot C_{5i})$$

3)

$$\textstyle\sum_i^n (A_{3i} \cdot \Im_{mxi} + B_{3i} \cdot \Im_{myi} + C_{3i} \cdot \Im_{mzi}) =$$

$$+K_1 \cdot [\textstyle\sum_i^n (A_{3i} \cdot A_{1i} + B_{3i} \cdot B_{1i} + C_{3i} \cdot C_{1i})]$$

$$+K_2 \cdot [\textstyle\sum_i^n (A_{3i} \cdot A_{2i} + B_{3i} \cdot B_{2i} + C_{3i} \cdot C_{2i})]$$

$$+K_3 \cdot [\textstyle\sum_i^n (A_{3i}^2 + B_{3i}^2 + C_{3i}^2)]$$

$$+K_4 \cdot [\textstyle\sum_i^n (A_{3i} \cdot A_{4i} + B_{3i} \cdot B_{4i} + C_{3i} \cdot C_{4i})]$$

$$+K_5 \cdot [\textstyle\sum_i^n (A_{3i} \cdot A_{5i} + B_{3i} \cdot B_{5i} + C_{3i} \cdot C_{5i})]$$

Simplifying,

$$Y_3 = K_1 \cdot X_{3,1} + K_2 \cdot X_{3,2} + K_3 \cdot X_{3,3} + K_4 \cdot X_{3,4} + K_5 \cdot X_{3,5} \tag{B.16}$$

where

$$\begin{aligned}
Y_3 &= \textstyle\sum_i^n (A_{3i} \cdot \Im_{mxi} + B_{3i} \cdot \Im_{myi} + C_{3i} \cdot \Im_{mzi}) \\
X_{3,1} &= \textstyle\sum_i^n (A_{3i} \cdot A_{1i} + B_{3i} \cdot B_{1i} + C_{3i} \cdot C_{1i}) \\
X_{3,2} &= \textstyle\sum_i^n (A_{3i} \cdot A_{2i} + B_{3i} \cdot B_{2i} + C_{3i} \cdot C_{2i}) \\
X_{3,3} &= \textstyle\sum_i^n (A_{3i}^2 + B_{3i}^2 + C_{3i}^2) \\
X_{3,4} &= \textstyle\sum_i^n (A_{3i} \cdot A_{4i} + B_{3i} \cdot B_{4i} + C_{3i} \cdot C_{4i}) \\
X_{3,5} &= \textstyle\sum_i^n (A_{3i} \cdot A_{5i} + B_{3i} \cdot B_{5i} + C_{3i} \cdot C_{5i})
\end{aligned}$$

4)

$$\begin{aligned}
&\textstyle\sum_i^n (A_{4i}\cdot \Im_{mxi} + B_{4i}\cdot \Im_{myi} + C_{4i}\cdot \Im_{mzi}) = \\
&+K_1 \cdot [\textstyle\sum_i^n (A_{4i}\cdot A_{1i} + B_{4i}\cdot B_{1i} + C_{4i}\cdot C_{1i})] \\
&+K_2 \cdot [\textstyle\sum_i^n (A_{4i}\cdot A_{2i} + B_{4i}\cdot B_{2i} + C_{4i}\cdot C_{2i})] \\
&+K_3 \cdot [\textstyle\sum_i^n (A_{4i}\cdot A_{3i} + B_{4i}\cdot B_{3i} + C_{4i}\cdot C_{3i})] \\
&+K_4 \cdot [\textstyle\sum_i^n (A_{4i}^2 + B_{4i}^2 + C_{4i}^2)] \\
&+K_5 \cdot [\textstyle\sum_i^n (A_{4i}\cdot A_{5i} + B_{4i}\cdot B_{5i} + C_{4i}\cdot C_{5i})]
\end{aligned}$$

Simplifying,

$$Y_4 = K_1 \cdot X_{4,1} + K_2 \cdot X_{4,2} + K_3 \cdot X_{4,3} + K_4 \cdot X_{4,4} + K_5 \cdot X_{4,5} \tag{B.17}$$

where

$$\begin{aligned}
Y_4 &= \textstyle\sum_i^n (A_{4i}\cdot \Im_{mxi} + B_{4i}\cdot \Im_{myi} + C_{4i}\cdot \Im_{mzi}) \\
X_{4,1} &= \textstyle\sum_i^n (A_{4i}\cdot A_{1i} + B_{4i}\cdot B_{1i} + C_{4i}\cdot C_{1i}) \\
X_{4,2} &= \textstyle\sum_i^n (A_{4i}\cdot A_{2i} + B_{4i}\cdot B_{2i} + C_{4i}\cdot C_{2i}) \\
X_{4,3} &= \textstyle\sum_i^n (A_{4i}\cdot A_{3i} + B_{4i}\cdot B_{3i} + C_{4i}\cdot C_{3i}) \\
X_{4,4} &= \textstyle\sum_i^n (A_{4i}^2 + B_{4i}^2 + C_{4i}^2) \\
X_{4,5} &= \textstyle\sum_i^n (A_{4i}\cdot A_{5i} + B_{4i}\cdot B_{5i} + C_{4i}\cdot C_{5i})
\end{aligned}$$

5)

$$\begin{aligned}
&\textstyle\sum_i^n (A_{5i} \cdot \Im_{mxi} + B_{5i} \cdot \Im_{myi} + C_{5i} \cdot \Im_{mzi}) = \\
&+K_1 \cdot [\textstyle\sum_i^n (A_{5i} \cdot A_{1i} + B_{5i} \cdot B_{1i} + C_{5i} \cdot C_{1i})] \\
&+K_2 \cdot [\textstyle\sum_i^n (A_{5i} \cdot A_{2i} + B_{5i} \cdot B_{2i} + C_{5i} \cdot C_{2i})] \\
&+K_3 \cdot [\textstyle\sum_i^n (A_{5i} \cdot A_{3i} + B_{5i} \cdot B_{3i} + C_{5i} \cdot C_{3i})] \\
&+K_4 \cdot [\textstyle\sum_i^n (A_{5i} \cdot A_{4i} + B_{5i} \cdot B_{4i} + C_{5i} \cdot C_{4i})] \\
&+K_5 \cdot [\textstyle\sum_i^n (A_{5i}^2 + B_{5i}^2 + C_{5i}^2)]
\end{aligned}$$

Simplifying,

$$Y_5 = K_1 \cdot X_{5,1} + K_2 \cdot X_{5,2} + K_3 \cdot X_{5,3} + K_4 \cdot X_{5,4} + K_5 \cdot X_{5,5} \tag{B.18}$$

where

$$\begin{aligned}
Y_5 &= \textstyle\sum_i^n (A_{5i} \cdot \Im_{mxi} + B_{5i} \cdot \Im_{myi} + C_{5i} \cdot \Im_{mzi}) \\
X_{5,1} &= \textstyle\sum_i^n (A_{5i} \cdot A_{1i} + B_{5i} \cdot B_{1i} + C_{5i} \cdot C_{1i}) \\
X_{5,2} &= \textstyle\sum_i^n (A_{5i} \cdot A_{2i} + B_{5i} \cdot B_{2i} + C_{5i} \cdot C_{2i}) \\
X_{5,3} &= \textstyle\sum_i^n (A_{5i} \cdot A_{3i} + B_{5i} \cdot B_{3i} + C_{5i} \cdot C_{3i}) \\
X_{5,4} &= \textstyle\sum_i^n (A_{5i} \cdot A_{4i} + B_{5i} \cdot B_{4i} + C_{5i} \cdot C_{4i}) \\
X_{5,5} &= \textstyle\sum_i^n (A_{5i}^2 + B_{5i}^2 + C_{5i}^2)
\end{aligned}$$

The above system of equations can be expressed in matrix form as:

$$\begin{bmatrix} X_{1,1} & X_{1,2} & X_{1,3} & X_{1,4} & X_{1,5} \\ X_{2,1} & X_{2,2} & X_{2,3} & X_{2,4} & X_{2,5} \\ X_{3,1} & X_{3,2} & X_{3,3} & X_{3,4} & X_{3,5} \\ X_{4,1} & X_{4,2} & X_{4,3} & X_{4,4} & X_{4,5} \\ X_{5,1} & X_{5,2} & X_{5,3} & X_{5,4} & X_{5,5} \end{bmatrix} \cdot \begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ Y_4 \\ Y_5 \end{bmatrix}$$

The solution is:

$$[K] = [X]^{-1} \cdot [Y]$$

Appendix C

Calibration Procedure II

This appendix gives the mathematical details for performing the multiple parameter linear regression analysis (LSE) to determine the refined calibration coefficients $\{a_1, a_2, a_3\}$.

It is assumed that each K_i can be represented in the form

$$K_i = e^{a_1 + a_2 V_j + a_3 \beta_j} \tag{C.1}$$

where

β_j : geometric constant (function of axial immersion)

V_j : average cutting velocity over cutting edge (function of axial immersion)

Taking the natural logarithm of the above equation for each K_i gives the form

$$\ln K_i = a_1 + a_2 \cdot V_j + a_3 \cdot \beta_j \tag{C.2}$$

For each K_i, over N tests, the parameters a_i are determined using multiple parameter log-linear regression (see appendix B for Normality assumptions)

$$SSE = \sum_{j}^{n} [\ln K_{i,j} - (a_1 + a_2 V_j + a_3 \beta_j)\,]^2$$

where

$i = 1, 2, 3, 4, 5$

β_j : geometric factor (function of axial immersion) at test j

V_j : average cutting velocity over cutting edge (function of axial immersion) at test j

The Sum Squared Error is minimized when

$$\frac{\delta SSE}{\delta a_i} = 0, \quad (i = 1, \ldots, 5)$$

the partial derivatives are given as:

$$\frac{\delta SSE}{\delta a_1} = -2 \cdot \sum_{j}^{n} \{\ln K_{i,j} - [a_1 + a_2 \cdot V_j + a_3 \cdot \beta_j]\,\} \tag{C.3}$$

$$\frac{\delta SSE}{\delta a_2} = -2 \cdot \sum_{j}^{n} \{\ln K_{i,j} \cdot V_j - [a_1 \cdot V_j + a_2 \cdot V_j^2 + a_3 \cdot (V_j \cdot \beta_j)]\,\} \tag{C.4}$$

$$\frac{\delta SSE}{\delta a_3} = -2 \cdot \sum_{j}^{n} \{\ln K_{i,j} \cdot \beta_j - [a_1 \cdot \beta_j + a_2 \cdot (\beta_j \cdot V_j) + a_3 \cdot \beta_j^2]\,\} \tag{C.5}$$

Recall that the Sum Squared Error is minimized when

$$\frac{\delta SSE}{\delta a_i} = 0, \quad (i = 1, \ldots, 5)$$

gives the following system of equations

1)

$$\sum_{j}^{n} \ln K_{i,j} = a_1 \cdot n + a_2 \sum_{j}^{n} V_j + a_3 \sum_{j}^{n} \beta_j \tag{C.6}$$

Simplifying.

$$Y_1 = a_1 \cdot X_{1,1} + a_2 \cdot X_{1,2} + a_3 \cdot X_{1,3}$$

where

$Y_1 = \sum_j^n \ln K_{i,j}$

$X_{1,1} = n$

$X_{1,2} = \sum_j^n V_j$

$X_{1,3} = \sum_j^n \beta_j$

2)

$$\sum_{j}^{n}(\ln K_{i,j} \cdot V_j) = a_1 \cdot \sum_{j}^{n} V_j + a_2 \sum_{j}^{n} V_j^2 + a_3 \sum_{j}^{n}(V_j \cdot \beta_j) \tag{C.7}$$

Simplifying,

$$Y_2 = a_1 \cdot X_{2,1} + a_2 \cdot X_{2,2} + a_3 \cdot X_{2,3}$$

where

$Y_2 = \sum_j^n (\ln K_{i,j} \cdot V_j)$

$X_{2,1} = \sum_j^n V_j$

$X_{2,2} = \sum_j^n V_j^2$

$X_{2,3} = \sum_j^n (V_j \cdot \beta_j)$

3)

$$\sum_{j}^{n}(\ln K_{i,j} \cdot \beta_j) = a_1 \cdot \sum_{j}^{n} \beta_j + a_2 \sum_{j}^{n}(\beta_j \cdot V_j) + a_3 \sum_{j}^{n} \beta_j^2 \tag{C.8}$$

Simplifying,

$$Y_3 = a_1 \cdot X_{3,1} + a_2 \cdot X_{3,2} + a_3 \cdot X_{3,3}$$

where

$Y_3 = \sum_j^n (\ln K_{i,j} \cdot \beta_j)$

$X_{3,1} = \sum_j^n \beta_j$

$X_{3,2} = \sum_j^n (\beta_j \cdot V_j)$

$X_{3,3} = \sum_j^n \beta_j^2$

The above system of equations can be expressed in matrix form as:

$$\begin{bmatrix} X_{1,1} & X_{1,2} & X_{1,3} \\ X_{2,1} & X_{2,2} & X_{2,3} \\ X_{3,1} & X_{3,2} & X_{3,3} \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix}$$

The solution is:

$[a] = [X]^{-1} \cdot [Y]$

Appendix D

Experimental Identification of Feed Drive Dynamics

D.1 Introduction

In CNC machining applications, the feed rate and the trajectory of the tool are typically specified in the G-Code file. The feed drive system implements the specified data and provides an actual feed rate and an actual tool path. Clearly, the actual cutting conditions and the resulting surface accuracy depend on the machine tool and its dynamic capabilities. To generate the tool motion accurately, the dynamics of the feed drive system should be integrated in the generation of the tool path.

Numerous publications have dealt with the identification and modeling of the dynamics of machine tool feed drive systems. The primary interest of these publications is to develop a feedforward tracking control strategy [50, 85, 88, 98, 38, 71, 86, 89, 90, 91, 47, 102], or cross-coupling control strategy [46, 51, 52, 48, 22, 49, 38, 47, 80] to compensate for the dynamic path errors. Although the accuracy of these controllers depends on the quality of the feed drive model, the important work done in this area has not emphasized the identification and modeling issues except the research done by Kulkarni [50] and Tung [90]. Review of the work presented by Kulkarni and Tung reveals that discrete-time models of feed drive dynamics may be obtained in numerous ways. The discrete time models may be determined analytically from the discrete-time equivalents of continuous-time models [17, 80, 4], or may be determined empirically by experimental process identification [50, 88, 51, 89, 90, 85, 80].

The analytical model design is based on exact knowledge of the system time constants, the system behavior and the order of the dead time of each machine axis. After setting up the transfer blocks for the subsystems, an S-Z transformation is carried out to obtain discretely sampled data. One of the problems with this method is that often there is insufficient knowledge about the individual subsystems. Consequently, the model can deviate from the current machine condition [98], causing the analytical solution not to fit the demands of the practical application.

The empirical method, based on experimental identification, offers a more exact modeling of the real feed drive dynamics. The effects such as amplifier nonlinearities and nonlinear friction components can be described implicitly by including the position loop in the identified model. It is also possible to obtain the current model structure parameters when the machine load is changed. The advantage of this method is that the transfer function model in the Z-domain can be obtained directly. For these reasons, the experimental feed drive modeling is favored. Kulkarni and Tung [50, 90] have proposed methodologies for experimental identification of the feed drive dynamics.

The identification methodology proposed by Tung [90] uses a standard least squares penalty function. The penalty function is transformed into the frequency domain, resulting in the frequency-weighted least squares method. The identification can be biased to emphasize the fit in the low frequency range. The bias is achieved by prefiltering the prediction errors, using a low-pass filter to manipulate the frequency weighting functions [91]. Obtaining a satisfactory bias distribution may involve trial and error. By prefiltering signals, the high frequency dynamics are removed. The filtered system is strongly susceptible to being over-modeled. It will also have an inverse term in the least squares solution that is rank-deficient, and will cause numerical instabilities. This method will not be applied here.

Two methodologies, composed of the pulse transfer function and the linear discrete-time model based on least squares identification, are described by Kulkarni [50]. The latter method predicts the feed drive behavior more accurately than the former. The appropriate discrete-time model is determined by the application of tests such as the response error test, the step response invariance test, the condition number test, the determinant ratio test, and the polynomial test. Details describing these tests can be found in [50]. These tests are not always conclusive in establishing the appropriate model order, mainly when they are applied separately [50].

The present study is based on a technique similar to the Kulkarni methodology. The Kulkarni method is modified to address the major problems of identification: the choice of an informative excitation, and the selection of the structure and orders of the time series model.

The remainder of this appendix is organized as follows. In section D.2, the automatic procedure for identifying the feed drive system model, based on time series modeling, is introduced. In section D.3, the experimental test facilities are described.

In section D.4, the characteristics of the excitation are specified to ensure the identification of a reliable model for a wide range of feed rates. In section D.5, the results of the identification are assessed. In section D.6, the experimental verification is described. In section D.7, the generation of the actual feed rate based on the obtained model is described.

D.2 Identification Procedure based on time series modeling

The feed drive system is assumed to be a linear time invariant system. This system is composed of a single input $u(t)$ and a single output $y(t)$. A general parametric model structure of such a system at time-instant $t = k \cdot T$, is expressed as [59]:

$$y(t) = G_p(z^{-1})u(t) + H(z^{-1})e(t) \tag{D.1}$$

$e(t)$ is the noise entering the system. z^{-1} is the one step delay operator. $G_p(z^{-1})$ and $H(z^{-1})$ are the plant and the noise transfer functions. They are defined as:

$$\begin{aligned} G_p(z^{-1}) &= \frac{z^{-n_k}B(z^{-1})}{A(z^{-1})F(z^{-1})} \\ H(z^{-1}) &= \frac{C(z^{-1})}{A(z^{-1})D(z^{-1})} \end{aligned} \tag{D.2}$$

n_k is the number of sample delays. The polynomials A, B, C, D, and F define the model structure. The orders of these polynomials are n_a, $n_b - 1$, n_c, n_d, and n_f, respectively. The transfer functions G_p and H can be obtained for a given model structure and order using the identification algorithms available in the "MATLAB system identification software package" [63]. The key issue is to identify the adequate model structure and the polynomial orders for a given set of Input/Output data.

The application of some relevant tests such as the response error test, the step response invariance test, the condition number test, the determinant ratio test, and the polynomial test in establishing the appropriate model order are not always conclusive. This is particularly true when they are applied separately [50].

One way to determine the appropriate model structure and orders is:

- to go through all the model structures
- to define a range of orders for each polynomial of the given structure
- to estimate the parametric model
- to evaluate specific criteria of the model adequacy
- to store the estimated values of these criteria in a matrix with the corresponding model structure and orders
- to choose the model structure and orders which lead to the best (user-defined) criteria of performance

The model adequacy is evaluated by examining the following performance criteria:

Criterion 1 cross-correlation between the input and the residuals: The absolute value of the cross-correlation between the residuals and the input signal, designated $\hat{\sigma}_{ue}$, must be less than the standard deviation σ_{ue} for the positive lags.

This ensures a 95% confidence interval for these variables [63].

$$\hat{\sigma}_{ue} \;\leq\; \sigma_{ue} \tag{D.3}$$

The hat $(\hat{.})$ stands for estimated value of variable $(.)$.

Criterion 2 whiteness of the residuals: The primary interest in this study is the dynamics of the plant $G_p(z^{-1})$. The correlation of residuals is accepted to exceed the standard deviation for few lags. However, the mean of the absolute value of the correlation of the residuals, designated $\hat{\bar{\sigma}}_e$, must be less than the standard deviation σ_e for lags greater than one.

$$\hat{\bar{\sigma}}_e \;\leq\; \sigma_e \tag{D.4}$$

Criterion 3 deviation of the frequency response function (FRF): The frequency response function of the time series model (FRFTM) is compared to the frequency response function (FRFPS) obtained using the power spectral method introduced by Welch [100]. The deviation of the amplitude $\hat{\sigma}_a$ and the phase $\hat{\sigma}_p$ can be expressed, in the frequency bandwidth of interest $[w_1 : w_n]$, as:

$$\hat{\sigma}_a = \frac{\sqrt{n}\cdot\sqrt{\sum_{i=1}^{n}\left(Ampl_{FRFTM}(w_i) - Ampl_{FRFPS}(w_i)\right)^2}}{\sum_{i=1}^{n} Ampl_{FRSPS}(w_i)} \tag{D.5}$$

$$\hat{\sigma}_p = \frac{\sqrt{n}\cdot\sqrt{\sum_{i=1}^{n}\left(Phase_{FRFTM}(w_i) - Phase_{FRFPS}(w_i)\right)^2}}{\sum_{i=1}^{n} Phase_{FRSPS}(w_i)} \tag{D.6}$$

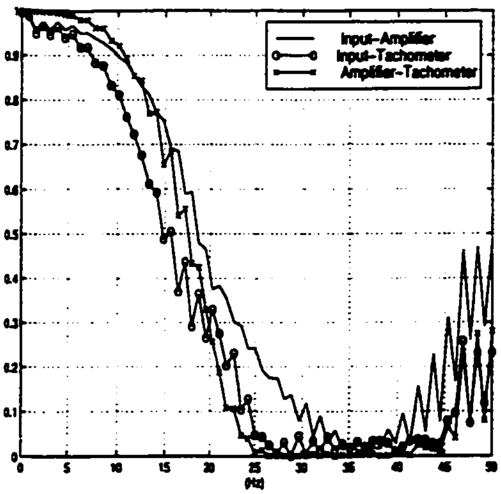


Figure D.1: Coherence between different signals

The deviations $\hat{\sigma}_a$ and $\hat{\sigma}_p$ should be less than the tolerance values σ_a and σ_p:

$$\begin{aligned} \hat{\sigma}_a &\leq \sigma_a \\ \hat{\sigma}_p &\leq \sigma_p \end{aligned} \tag{D.7}$$

In machine tools, the frequency spectrum of the reference trajectory is composed of low frequency signals [90]. The identification procedure should, therefore, focus on the low frequency range. The system can be considered linear when the coherence function is greater than 0.8. The examination of the coherence function between different signals obtained from the feed drive of this study shows that the frequency bandwidth of interest is 0-10 (Hz) (figure D.1).

Criterion 4 the simulation fit: A fresh set of data is used to perform a simulation of the parametric model. The simulation fit is determined by the expression of the standard deviation , which must be less than a tolerance value :

$$(\hat{\sigma}_f = \sqrt{\sum_{Points} \frac{(\hat{y}-y)^2}{NumberPoints}}) \leq \sigma_f \tag{D.8}$$

Criterion 5 zeros and poles cancellation: In numerical control, it is common to have some zeros outside the unit circle [11]. However, the poles of the plant model G_p should be within the unit circle.

$$\hat{\sigma}_{ZP} = max\,(|Poles|) \leq 1 \tag{D.9}$$

In addition, it must be noted that an oscillatory behavior may happen at one half the Nyquest frequency for roots in the left half of the Z-plane, that is roots with negative real parts. In a system under computer control, roots in the left half of Z-plane can represent oscillation forced by the controller [13]. Then, the criteria is that the real part of the roots should be positive:

$$\hat{\sigma}_{P+} = real\,(Poles) \geq 0 \tag{D.10}$$

Criterion 6 the standard deviation of poles and zeros: The model uncertainty is manifested in the standard deviations of the pole(s) and zero(s) locations. Large uncertainties are caused by excessively high model orders, by inadequate excitation, or by low signal-to-noise ratios [63]. Since the criterion (1) selects the models for which the signal-to-noise ratio is large, high standard deviations of

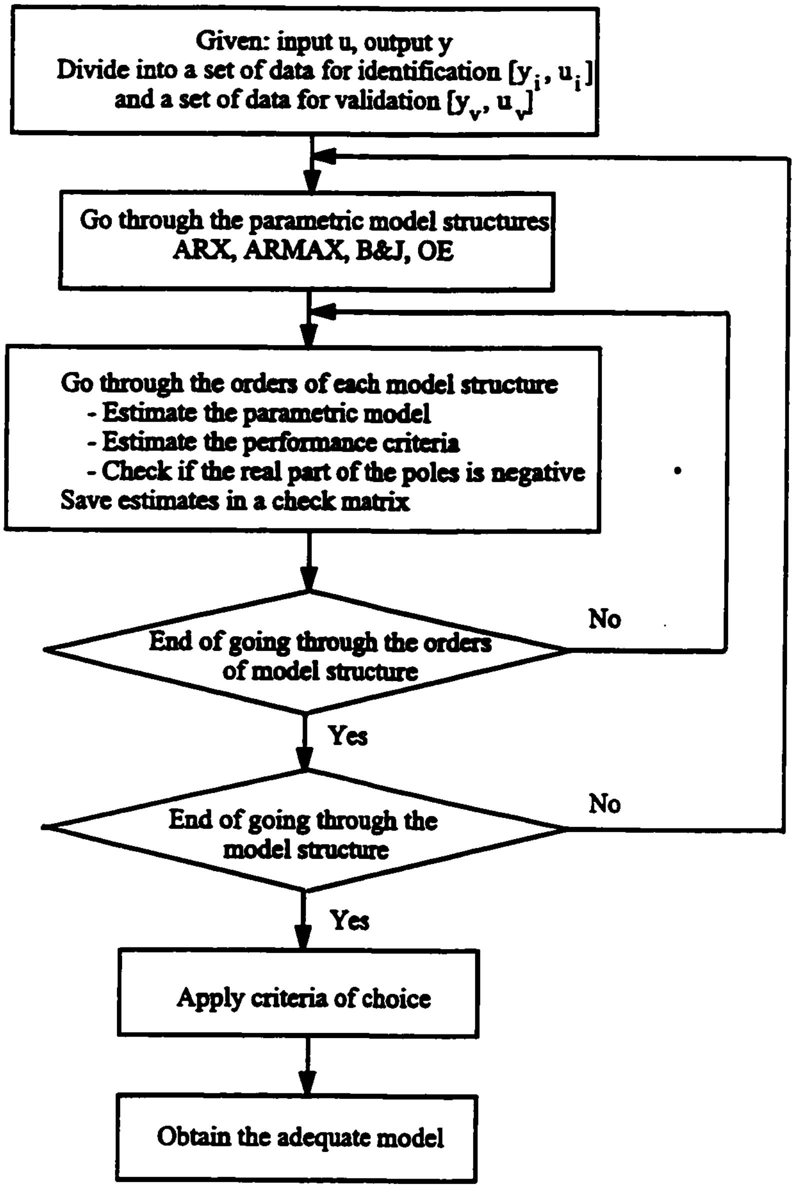


Figure D.2: Flowchart for the selection of the appropriate model structure and order the pole(s) and zero(s) locations will indicate excessively high model orders. The standard deviation of the roots and zeros should be less than certain tolerance values σ_{ZZL} and σ_{ZPL}:

$$(\hat{\sigma}_{ZZL} = max\left(\frac{|Error\ on\ the\ Zero|}{|Zero|}\right)) \leq \sigma_{ZZL} \qquad (D.11)$$

$$(\hat{\sigma}_{ZPL} = max\left(\frac{|Error\ on\ the\ Pole|}{|Pole|}\right)) \leq \sigma_{ZPL} \qquad (D.12)$$

This procedure is summarized by the flowchart depicted in figure D.2.

D.3 Experimental set up

The experiments are carried out on a 5-axis vertical milling machine "FADAL VMC4020". The feed drive system used in this study is the X-axis unit. The feed drive is composed of a permanent dc motor coupled to the ball screw that transmits a linear motion to the XY bed of the machine through the ball screw nut. The XY bed is constrained to straight-line motion using linear slides. The travel length is limited to 40 inches for the X-axis. The motor is regulated by means of a pulse width modulated amplifier. The feed drive control system has two cascaded control loops (figure D.3):

- the angular velocity control loop that includes a tachometer providing the velocity feedback signal
- the position control loop that includes a resolver providing the actual position feedback signal. The resolver is coupled to the motor shaft.

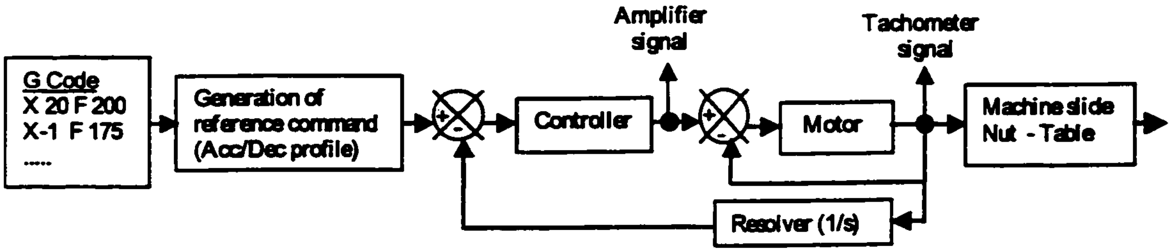


Figure D.3: Feed drive system schema

The numerical control unit "FANUC CNC88" interpolates the desired path defined by the user specified G-Code file and calculates the trajectory based on an acceleration/deceleration profile (Acc/Dec). The trajectory is then supplied as a reference command signal for the position control loop. The numerical control unit also performs specific discrete logic control for the miscellaneous machine functions, (i.e. spindle on, coolant on, etc.).

The G-Code file is generated on a personal computer (PC), and then transferred to the CNC control unit via an RS-232 serial connection. Each G-Code file is composed of three parts: header, body, and footer. The header specifies the general set up of the machine. In this study, the header includes the interpolation plane (XY plane (G17)), the interpolation mode (linear (G01)), the position unit (inch (G20)), the reference mode (incremental (G91)), and the blend mode (with no return to zero feed rate at each change of feed rate (G08)). The function (G08) ensures a continuity of the velocity profile and a higher productivity while machining. The function (M49) is introduced to disable the feed/speed operator override. To synchronize data acquisition with the start of program execution, two M functions are introduced at the end of the header. These functions are (M67) and (M66). The falling edge of the (M67)/(M66) signal will be used as an external trigger for the data acquisition system. The acquisition will start at the same time as the processing of the trajectory blocks. The body is composed of the trajectory specifications. The footer specifies the return to the home position and the end of the program.

The command signal and the actual angular velocity of the motor are sensed from the amplifier and the tachometer. A voltage divider circuit with ground isolation is used to condition the signals. The command signal, the tachometer signal, and the external trigger are supplied to the PC through the data acquisition card (ADC). A set of software routines was implemented on the PC to monitor the ADC card and perform data acquisition.

The sampling frequency should always be less than 250 Hz [98, 85]. When higher sampling frequencies are used, high order effects such as amplifier dynamics will contaminate the identification [90, 91]. The reference trajectory is composed of low frequency signals. A sampling frequency of 100 Hz is sufficient to study phenomena up to a cut off frequency of 40 Hz. This sampling frequency will be adopted throughout this study.

D.4 Characteristics of the excitation

To identify the feed drive system, the input should be persistently exciting to generate data that are sufficiently informative. Two characteristics of input excitation that need to be considered are the second order properties of the input, and the shape of the signal to realize a given input spectrum.

The second order properties of the input include power spectrum and cross spectrum between input and driving noise [15, 59, 68]. This means that the input spectrum should have a frequency bandwidth that matches the frequency response of the process. The input should not be correlated with noise and the signal-to-noise ratio should be high. The variance of the parameter estimates depends only on these properties. Since the trajectory profile contains low frequency signals, the input should have dominant low frequency components. This will avoid exciting higher frequency modes of the feed drive system.

The second characteristic is the shape of the signal to produce a given input spectrum [15, 59, 68]. If the input moves have short time duration, they contain excitation energy at high frequencies but only a small amount of energy at low frequencies. For input moves having large time duration, these contain excitation energy at low frequencies but only a small amount of energy at high frequencies. A step input, for instance, is a low frequency excitation. In practice, it is often difficult to obtain enough detail and resolution in the recorded step response to determine the exact dynamics of the process. All the dynamic components except the integration are lumped into the dead time term [13]. To validate the time series model, an estimation of the frequency response function of the system is needed. The response of a step input does not contain enough information to allow smoothing of spectral estimates. To obtain this spectral estimate, the observed data is divided into sections to make an average estimation of the frequency response function. A linear model of the overall feed drive dynamics is acceptable because of the linearizing effect of the feedback in the velocity and position loop. However, there is no guarantee that a model obtained with given amplitude of the step input is reliable for another step input. Therefore, the input should have varying amplitude. Pandit [67] suggests that least squares minimization of the equation error is used to determine an equivalent linear model of the feed drive behavior. The linear model should then be valid in the neighborhood of the trajectory used for identification.

To satisfy the above constraints, a particular input is shaped. A multilevel sequence of the feed rate is used for identification. With this type of input, lower frequency components predominate [66]. The input sequence is controlled by three basic parameters: the feed rate range, the number of feed rate levels and the period of travel for a given feed rate amplitude. These parameters are carefully considered to avoid a nonlinear response due to saturation.

Suppose that the feed rate range is $[-F_{max}, F_{max}]$, where F_{max} is a given positive value of the feed rate. This range is devised into N equal intervals ΔF, $\Delta F = \frac{2F_{max}}{N}$, producing $N + 1$ levels F_i of the Feed rate:

$$F_i = -F_{max} + i \cdot \Delta F \quad i = 0:1:N \tag{D.13}$$

N is chosen to be odd to avoid a zero value for the feed rate. A path is then defined with successive feed rate steps. A psuedo-random binary sequence of feed rate steps is used. The total path length is an additional constraint imposed to ensure that the travel does not exceed the limits of the machine. Each feed rate step starts from one level F_i of the feed rate and has an amplitude equal to a multiple of the feed rate increment ΔF. All the combinations of the allowed feed rate levels F_i and feed rate jumps are used. Figure D.4 depicts a result of feed rate profile obtained with $F_{max} = 175$ *ipm* and $N = 7$.

The period of each feed rate step is made long enough so that the feed drive unit

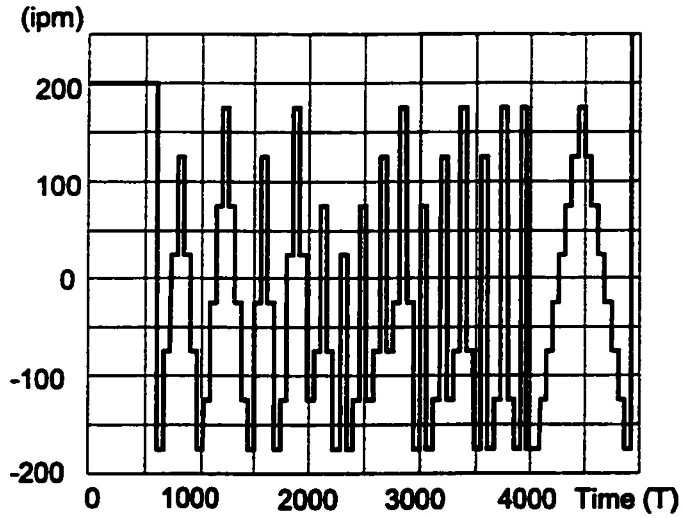


Figure D.4: Input signal

can achieve the feed jump without interfering with the next feed rate step.

This input ensures the reliability of the model over a wide range of feed rates. The feed rates are chosen to be within the limits of the feed drive system. The system model gained by this input sequence is the average of the forward and backward models for the machine axis because the input sequence is composed of negative and positive feed rate.

D.5 Identification experiments and results

The objective is to identify the transfer function relating the actual feed rate of the table to the commanded feed rate specified by the G-Code file. This transfer function will include the controller interpolator as well as the dynamics of the feed drive mechanism. The input of the system is then the commanded feed rate and the output is the tachometer signal. In this study, the actual feed rate of the table is supposed proportional to the tachometer signal. The dynamics of the slides and ball screw are excluded.

A G-Code file containing the excitation defined previously is executed on the machine. The observation period is 55 seconds. Using the sampling frequency of 100Hz, the amplifier and tachometer signals are composed of 5500 points. A comparison of the feed rate signal obtained by sampling the commanded feed rate and the tachometer signal reveals that the period of the commanded feed rate is less than the period of the tachometer signal. This is caused by two main factors. The first is that to achieve a feed rate jump instantaneously requires the feed drive to have infinite acceleration or deceleration. The actual system needs a period of time to accelerate or decelerate. The second factor is friction. Friction affects the performance of the feed drive system by increasing the period of the travel. For a constant reference command, the position error reaches a steady state when the axis moves with a constant velocity. When the position counter reaches the final required position, the flow of the reference pulses is blocked. However, because the axis is still in motion, the feedback pulses are sent to reduce the content of an up-down counter to zero an instant later. This procedure continues until the actual position is equal to the required one and the tool is at the end of the segment.

In this study, the primary concern is to model the actual Acc/Dec profile of the feed drive system. The contribution of friction effects in the expansion of travel time will not be included in the modeling. If a feed rate set is defined, the feed drive system will accelerate to a certain value, then decelerate to achieve the feed rate jump and reach the desired feed rate smoothly. At a steady state, the time constants of the feed

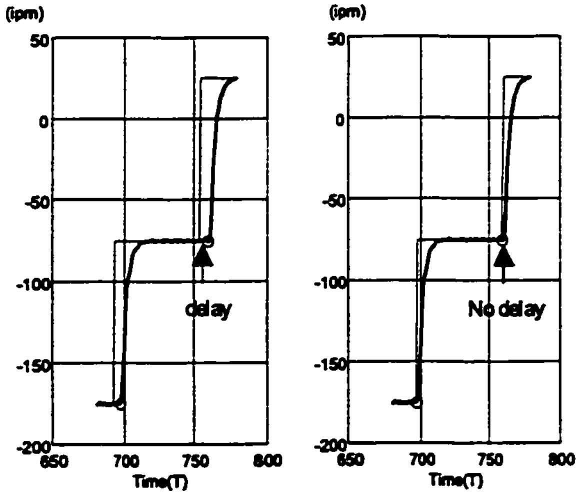


Figure D.5: Expansion in time of the input signal; Input signal (thin line), Tachometer signal (thick line)

drive system do not affect the system performance and the transfer function is reduced to a constant. The transient part of the resulting tachometer signal is a signature of the Acc/Dec profile. The interest here is the transient phase of the tachometer signal. The feed rate defined by the G-Code file is expanded in time so that the tachometer signal becomes causal with no delay in response to each commanded feed jump as depicted in figure D.5.

Each signal is divided into two parts. The first part has a length of 4096 points and will be used for identification purposes. The second part has a length of 768 points and directly follows the end point of the first part. The second part is used for cross validation purposes. The identification procedure introduced in Section D.2 is then applied. The identification procedure is time consuming if no bounds are put on polynomial order. To decrease the computing time, the range of the polynomial

Polynomial order	n_a	n_b	n_c	n_d	n_f	n_k
Minimum value	2	1	1	1	2	1
Maximum value	4	4	4	4	4	3

Table D.1: Range of polynomial orders

σ_{ue}	σ_e	σ_a	σ_p	σ_f	σ_{ZP}	σ_{ZZL}	σ_{ZPL}
1	1	0.1	0.1	0.3	1	0.1	0.1

Table D.2: Tolerance values of the performance criteria

order is limited based on insight into the physical system and/or based on the results of previous research efforts. The system should be causal. Therefore, the following inequalities should be respected:

$$n_b \leq n_a \; ; \; n_b \leq n_f \; ; \; n_c \leq n_a \; ; \; n_c \leq n_d \tag{D.14}$$

In addition, previous researchers have generally modeled the plant with a second or third order system. The search area is increased here to include a range of orders given in table D.1. The tolerance values for the criteria of selecting the adequate model are specified in table D.2. According to these tolerance values, a set of four models is obtained. To choose one of these models, the criterion of minimum fit error is applied. The resulting model is an ARMAX model with the following orders: $n_a = 3$, $n_b = 1$, $n_c = 1$, and $n_k = 1$. This model is expressed by the equation:

$$A(z^{-1})y(t) = B(z^{-1}) \cdot u(t - n_k \cdot T) + C(z^{-1})e(t) \tag{D.15}$$

		(Z^0)	(Z^{-1})	(Z^{-2})	(Z^{-3})
$B(z^{-1})$	Coefficient	0.1411			
	Deviation	0.0000			
$A(z^{-1})$	Coefficient	1	-1.4978	0.8782	-0.2386
	Deviation	0	0.0133	0.0199	0.0084
$C(z^{-1})$	Coefficient	1	-0.5568		
	Deviation	0	0.0138		

Table D.3: The polynomial coefficients and their deviation

The polynomial coefficients and their deviation are given in table D.3. The deviation of the polynomial coefficients is less than 5%. This model can be written as:

$$\begin{aligned} y(t) &= \frac{0.1411Z^{-1}}{1-1.4978Z^{-1}+0.8782Z^{-2}-0.2386Z^{-3}}u(t) \\ &-\frac{30.67\,(1-0.5568Z^{-1})}{1-1.4978Z^{-1}+0.8782Z^{-2}-0.2386Z^{-3}}e(t) \end{aligned} \tag{D.16}$$

where $u(t)$ corresponds to the commanded feed rate in (ipm). $y(t)$ corresponds to the actual feed rate of the table in (ipm).

The transfer function G_p is then defined by:

$$G_p(z) = \frac{0.1411Z^{-1}}{1-1.4978Z^{-1}+0.8782Z^{-2}-0.2386Z^{-3}} \tag{D.17}$$

This transfer function has two complex poles and a real pole. These poles and their standard deviation are given in table D.4. The deviation of these poles is very small.

The estimated values of the performance criteria for this model are given in table D.5.

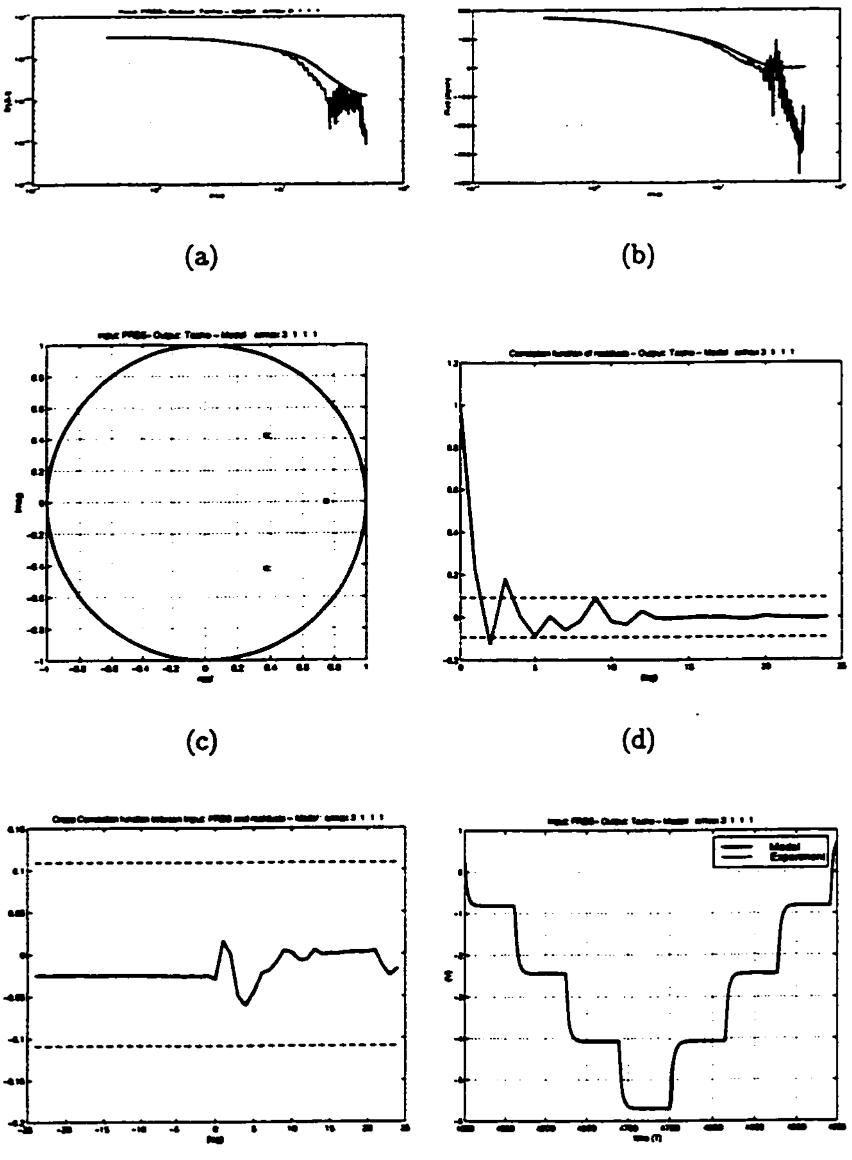
Poles	0.7519	0.3729+0.422j	0.3729-0.422j
Deviation	0.0022	0.0063+0.0085j	0.0063+0.0085j

Table D.4: The poles of the plant transfer function and their deviation

$\hat{\sigma}_{ue}$	$\overline{\hat{\sigma}}_e$	$\hat{\sigma}_a$	$\hat{\sigma}_p$	$\hat{\sigma}_f$	$\hat{\sigma}_{ZP}$	$\hat{\sigma}_{ZZL}$	$\hat{\sigma}_{ZPL}$
0.412	0.557	0.042	0.059	0.260	0.752	0	0.019

Table D.5: Estimated values of the performance criteria

The values of the performance criteria express the adequacy of the model. This adequacy can be visualized better from the plots of the model amplitude and phase compared to the frequency response function obtained by spectral analysis (figures D.6a, and D.6b). It should be noted that the least square identification methods heavily penalize the high frequency miss-fit. Since our concern is the low frequency components, the actual response is largely in harmony with the model response. In addition, all the poles are located in the right half of the Z-plane within the unit circle (figure D.6c). The auto-correlation of residuals is within the standard deviation range for almost all the lags ensuring the whiteness of noise (figure D.6d). The cross-correlation between the input and residuals are within the standard deviation limit ensuring the 95% confidence interval for the model (figure D.6e). Finally, the simulation of the model with a fresh set of data that is not used for the identification shows the adequacy of the model in time domain (figure D.6f). This model is accepted and used for further development.



(a) (b)

(c) (d)

(e) (f)

Figure D.6: (a) Amplitude Plot of the model (thick line) together with spectral analysis estimate (thin line). (b) Phase Plot of the model (thick line) together with spectral analysis estimate (thin line). (c) Zeros (O mark) and Poles (X mark) cancellation. (d) Correlation of residuals. Horizontal bars indicate 95(e) Cross-correlation between input and residuals. Horizontal bars indicate 95(f) Simulated output of the model (thick line) compared to the measured output (thin line).

D.6 Experimental verification

In the previous section, the transfer function of the feed drive system was identified. Transforming this transfer function into the time domain provides a time model of the Acc/Dec profile of the feed drive system. An experimental validation of this model is proposed in this section.

D.6.1 Time model of the Acc/Dec profile

The transfer function of the feed drive system is composed of two complex poles and one real pole and can be expressed as:

$$G_p(z) = \frac{0.1411Z^{-1}}{[1-0.7519Z^{-1}]\,[1-(0.3729+0.422j)\,Z^{-1}]\,[1-(0.3729-0.422j)\,Z^{-1}]} \tag{D.18}$$

Combining the two complex poles leads to:

$$G_p(z) = \frac{0.1411Z^{-1}}{[1-0.7519Z^{-1}]\,[1-0.0.7458Z^{-1}+0.3127Z^{-2}]} \tag{D.19}$$

Equation (D.19) can be written in the standard format:

$$G_p(z) = \frac{b_0 z^{-1}}{[1-\exp^{-cT} z^{-1}]\cdot[1-2\exp^{-aT}\cos(bT)z^{-1}+\exp^{-2aT} z^{-2}]} \tag{D.20}$$

where $b_0 = 0.1411$, $a = 57.24$, $b = 8471$, and $c = 28.515$.

To determine the time representation of the feed drive model, the expression of the feed drive model given by equation (D.20) is first expanded into partial fractions:

$$\begin{aligned} G_p(z) &= \frac{ZG_1}{(1-\exp^{-cT} z^{-1})} \\ &+ \frac{ZG_2\left(1-\exp^{-aT}\cos(bT)z^{-1}\right)+ZG_3\exp^{-aT}\sin(bT)z^{-1}}{(1-2\exp^{-aT}\cos(bT)z^{-1}+\exp^{-2aT}z^{-2})} \end{aligned} \tag{D.21}$$

where:

$$\begin{aligned} ZG_1 &= \frac{b_0\exp^{-cT}}{\exp^{-2aT}+\exp^{-2cT}-2\exp^{-aT}\exp^{-cT}\cos(bT)} \\ ZG_2 &= -ZG_1 \\ ZG_3 &= \frac{\exp^{-2aT}-\exp^{-aT}\exp^{-cT}\cos(bT)}{\exp^{-aT}\exp^{-cT}\sin(bT)}\cdot ZG_1 \end{aligned} \tag{D.22}$$

The numerical values of these parameters are: $ZG_1 = 0.33$, $ZG_2 = -0.33$, $ZG_3 = 0.037$.

Applying the s transformation to this equation gives:

$$G_p(s) = \frac{ZG_1}{s+c} + \frac{ZG_2\,(s+a)+ZG_3 b}{(s+a)^2+b^2} \tag{D.23}$$

The velocity response, F_a, of the feed drive to a commanded feed rate, F_S, is given by the relationship:

$$F_a(s) = G_p(s)\cdot F_S(s) \tag{D.24}$$

Using equation (D.24), the Acc/Dec profile is defined, for a given step of the feed rate ΔF, by:

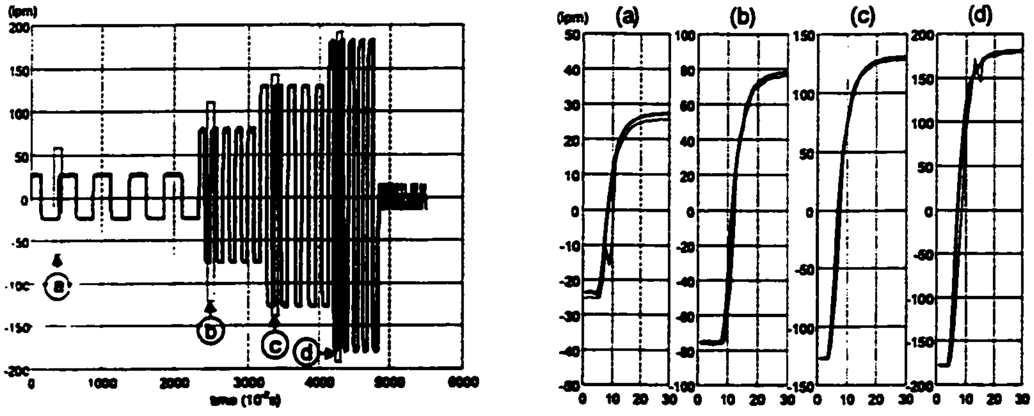
$$AccDec(s) = G_p(s) \cdot \Delta F \tag{D.25}$$

The time model of the Acc/Dec profile is expressed by:

$$\begin{aligned} AccDec(t) &= \big[ZG_1 \exp^{-ct} \\ &\quad + \exp^{-at}\left(ZG_2 \cos(bt) + ZG_3 \sin(bt)\right)\big] \cdot \Delta F \end{aligned} \tag{D.26}$$

D.6.2 Validation

An experiment was conducted to observe the variation of the Acc/Dec profile for different feed rate steps. In this experiment, the feed drive was commanded to change the feed rate between two opposite levels: -25 to 25ipm, -75 to 75 ipm, -125 to 125ipm, and -175 to 175ipm. This experiment provides different feed rate steps equal to 50, 150, 250, and 350 ipm (figure D.7a). For each step of the feed rate, a number of cycles are considered to provide data to evaluate an average Acc/Dec profile. The amplifier and the tachometer signals were recorded and are depicted in figure D.7a with thick and thin lines, respectively. The period of each move is set long enough so that the feed drive system achieves the feed rate step without interfering with the next move. The feed rate profile resulting from each step is observed by zooming in on the experimental data in the zones (a), (b), (c), and (d), respectively. The results are depicted in figure D.7b. It is clearly observed that the feed rate profiles are similar during the transient phase of the different feed rate steps. The tachometer signals exhibit some high frequency components. The high frequency components



(a)

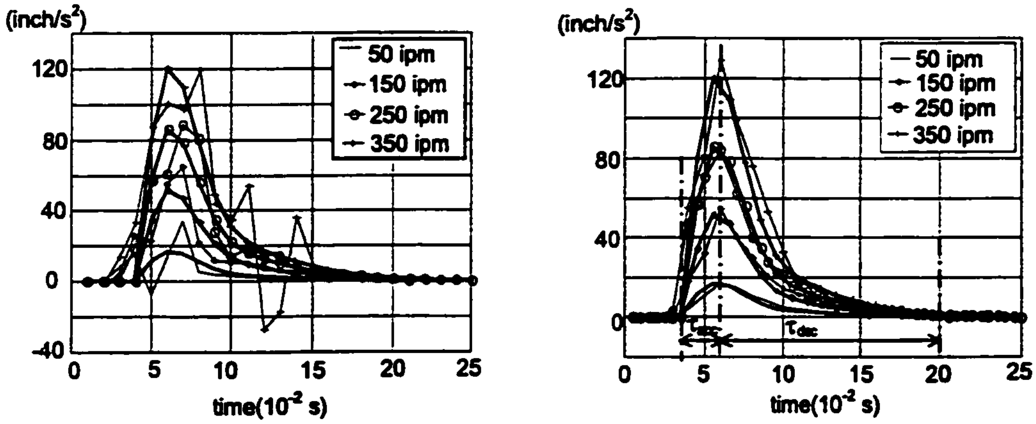
(b)

Figure D.7: Amplifier and tachometer signals: Amplifier (thick lines), Tachometer (thin lines)

are observed better when the derivative, with respect to time, is evaluated for the tachometer signals.

Taking the derivative with respect to time, the amplifier and the tachometer signals provide the Acc/Dec profiles for each feed rate jump. The validity of the Acc/Dec profile is examined by comparing the experimental results to theoretical estimates obtained from equation (D.26). Experimental results based on the tachometer signals (thin lines) are plotted against the theoretical estimations (thick lines) in figure D.8a. Experimental results based on the amplifier signals (thin lines) are plotted against the theoretical estimations (thick lines) in figure D.8b.

Figures D.8a and D.8b show that the magnitude of the Acc/Dec profile for a given feed rate step is proportional to the amplitude of this step. Figures D.8a and D.8b also show that the Acc/Dec profiles are composed of two phases. The first is the



(a)

(b)

Figure D.8: Acc/Dec profiles for different feed jump: experimental with thin lines, predicted with thick lines

acceleration phase where the acceleration increases from zero to a maximum value proportional to the feed rate jump. The periods of the acceleration phase are approximately identical, independent of the amplitude of the feed rate step. This period, denoted τ_{acc}, is equal to the time when the Acc/Dec profile reaches its maximum. τ_{acc} is a time constant. The second phase is the deceleration phase where the acceleration decreases from the maximum value to a value where the deceleration is considered finished. This phase allows the feed drive to reach the commanded feed rate smoothly. The periods for the deceleration phase, designated τ_{dec}, are also approximately identical, independent of the amplitude of the feed rate step. τ_{dec} is a time constant. The time constant τ_{total} is defined as follows:

$$\tau_{total} = \tau_{acc} + \tau_{dec} \tag{D.27}$$

All these time constants are determined in the next section. The good agreement between both experimental and simulated Acc/Dec time constants, and feed rate jump dependant magnitudes, validates the model.

A discrepancy was observed between the Acc/Dec model and the Acc/Dec profiles obtained based on the tachometer signals. This discrepancy is due to the presence of high frequency components in the tachometer signals. The friction of the slides and the disturbances due to the motion of the table are reflected back on the tachometer. The tachometer signals are, therefore, contaminated with these nonlinearities and exhibit high frequency components (figure D.8a). This discrepancy disappears when the Acc/Dec model is compared to the Acc/Dec profiles obtained based on the amplifier signals (figure D.8b). The amplifier signals can be considered as obtained by pre-filtering the high frequency components of the tachometer signals. The good agreement between the Acc/Dec model and the Acc/Dec profiles based on the amplifier signals proves the validity of the model for studying low frequency phenomena in the feed drive system. However, the nonlinearities that are not modeled, such as the friction and the mechanical dynamics of the table assembly, act to slow down the tool motion. The actual period of the tool motion is, therefore, greater than predicted.

D.7 Acc/Dec processor based on FIR filter

The objective of this part is to accurately depict the motion profile considering the limitation of the feed drive acceleration. The model identified previously for the feed drive system will be used as an Acc/Dec processor to generate the actual feed rate, F_a, from the commanded feed rate, F_S, obtained by sampling the feed rate specified by the G-Code file. The velocity response of the axis to a commanded feed rate is given, in the Z-domain, by:

$$\begin{aligned} F_a(z) &= G_p(z)F_S(z) \\ &= \frac{b_0 z^{-1}}{[1-\exp^{-cT} z^{-1}]\cdot[1-2\exp^{-aT}\cos(bT)z^{-1}+\exp^{-2aT} z^{-2}]} F_S(z) \end{aligned} \tag{D.28}$$

Chen [20] has demonstrated that a low pass discrete time FIR filter is a good Acc/Dec profile generator. Let $F_a(kT)$ be manipulated as a weighted moving average of the recent N_c commanded feed rate, $F_S(iT)$, $i \in [k : k - N_c - 1]$, associated with the appropriate weighted coefficients g_i, $i \in [0 : N_c - 1]$. The input-output feed rate relationship are expressed by:

$$F_a(k\cdot T) = \frac{1}{g_s}\sum_{i=0}^{N_c-1} g_i \cdot F_S\left((k-i)\cdot T\right) \tag{D.29}$$

Applying the Z transformation to equation (D.29) results in:

$$F_a(z) = \left(\frac{1}{g_s}\sum_{i=0}^{N_c-1} g_i \cdot z^{-i}\right)\cdot F_S(z) \tag{D.30}$$

where N_c is the order of the filter and g_s is a convergent factor. The actual acceleration $AccDec(kT)$ is defined by:

$$AccDec(kT) = \frac{F_a(kT) - F_a\left((k-1)\cdot T\right)}{T} \tag{D.31}$$

Introducing the expression of the actual feed rate given by equation (D.29) results in:

$$\begin{aligned} AccDec(kT) &= \frac{1}{Tg_s}\left(\sum_{i=0}^{N_c-1} g_i F_S\left((k-i)\cdot T\right)\right. \\ &\qquad \left. - \sum_{i=0}^{N_c-1} g_i F_S\left((k-i-1)\cdot T\right)\right) \end{aligned} \tag{D.32}$$

For a step input, for instance $F_S(kT) = \Delta F$ for $k \in [1 : N_c]$ and $F_S(0) = 0$, equation (D.32) will be:

$$AccDec(kT) = \frac{g_{k-1}}{g_s} \cdot \frac{\Delta F}{T} \tag{D.33}$$

This equation demonstrates that the Acc/Dec profile is defined by the filter coefficients g_k. The behavior of the Acc/Dec processor is then equivalent to a dynamic system of order $N_c - 1$ defined by the FIR filter coefficients C_k:

$$C_k = \frac{g_{k-1}}{g_s} \quad k \in [1 : N_c] \tag{D.34}$$

The feed drive model should, therefore, be transformed into an equivalent FIR filter. This transformation is obtained using long division. However, the use of the resulting filter is limited to signals that are sampled at the sampling frequency used for the identification of the feed drive model. One method that can be employed to solve this problem is to use the time representation of the feed drive model described by equation (D.26).

The FIR filter will correspond to the values of this Acc/Dec profile at a given sampling frequency. The feed rate resulting from the Acc/Dec processing can be made to converge to the desired feed rate for a steady state by normalizing the FIR

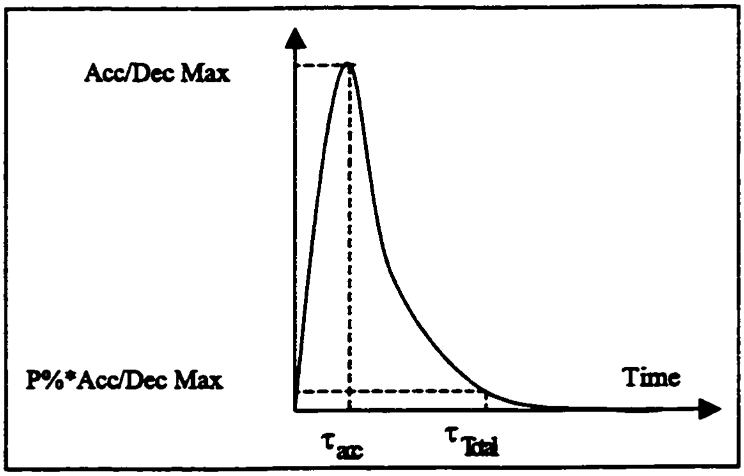


Figure D.9: Characteristics of the Acc/Dec profile

filter coefficients. The coefficient g_s is equal to the sum of coefficients g_k:

$$g_s = \sum_{k=0}^{N_c-1} g_k \tag{D.35}$$

However, the order or the number of the filter coefficients, N_c, is still unknown. The order of the filter is defined as the integer part of the quotient of the Acc/Dec profile time constant, τ_{total}, divided by the sampling period.

$$N_c = Int\left(\frac{\tau_{total}}{T}\right) \tag{D.36}$$

τ_{total} is the total period for the Acc/Dec processing. It is assumed that the Acc/Dec processing is completed at a given percentage P%, for instance P%=0.01, of the maximum value of the Acc/Dec profile (figure D.9). Two parameters must be determined: the instant τ_{acc} at which the acceleration is maximum and τ_{total} (figure D.9). The parameter τ_{acc} and τ_{total} are determined by solving the following nonlinear equations:

$$\begin{aligned}\exp^{-a\tau_{acc}}(ZG_3 b - ZG_2 a)\cos(b\tau_{acc}) &= \exp^{-a\tau_{acc}}(ZG_3 a - ZG_2 b) \\ &\quad \sin(b\tau_{acc}) + ZG_1 c \exp^{-c\tau_{acc}} \qquad \text{(D.37)}\end{aligned}$$

$$\begin{aligned}&\left(\exp^{-c\tau_{total}} - P\% \cdot \exp^{-c\tau_{acc}}\right) + \left(\exp^{-a\tau_{total}} - P\% \cdot \exp^{-a\tau_{acc}}\right) \\ &\qquad \cdot \left(\frac{ZG_3}{ZG_1}\sin(b\tau_{total}) - \cos(b\tau_{total})\right) = 0 \qquad \text{(D.38)}\end{aligned}$$

Solving these equations gives: τ_{acc}=0.022s and τ_{total}=0.177s. Once the filter is designed, equations (D.30), (D.31), (D.33), and (D.35) are used to generate the actual feed rate from the commanded feed rate according to the formula:

$$F_a(z) = \left(\sum_{i=1}^{N_c} C_j \cdot z^{1-i}\right) \cdot F_S(z) \qquad \text{(D.39)}$$

This procedure allows the actual feed rate to be evaluated quickly. However, the efficiency of this procedure is reduced when there is a direction reverse in the motion resulting in path and contouring errors. This study deals with the single axis motion, the contouring problem is beyond the scope of this work.

D.8 Summary

The objective of this study is to identify the low frequency dynamics of the feed drive system. This study is limited to the identification of the transfer function relating the actual feed rate, assumed to be proportional to the motor shaft velocity, to the desired feed rate specified in the G-Code file.

An empirical method based on experimental identification has been adopted because it results in a more accurate model for the real feed drive dynamics.

A systematic procedure of identification based on time series modeling is developed. In this procedure, specific criteria are introduced to evaluate the adequacy of the model and to select the appropriate model structure and orders.

The input excitation used for identification is considered to account for both the spectrum and the shape. The input excitation is specified as a multilevel sequence of feed rate with mainly low frequency components. The input is shaped so that the identified model is valid over a wide range of the feed rate within the physical limits of the machine. The identified model is an average of the forward and backward models of the feed drive system.

The application of the identification procedure results in an ARMAX model with the following orders: $n_a = 3$, $n_b = 1$, $N_c = 1$, and $n_k = 1$. The adequacy of this model has been validated in the frequency domain as well as in the time domain.

This model is transformed into a FIR filter and used as an Acc/Dec processor to generate the actual feed rate from the desired feed rate specified by the G-Code file. The transformation considers variable, user-specified sampling frequencies different from that used for identification. The FIR filter coefficients have been normalized so that the actual feed rate converges to the desired feed rate at a steady state.

The application of the Acc/Dec processor only describes the acceleration or the deceleration of the feed drive. It does not account for the effects of friction. These effects make the period of travel longer than the period of the feed rate resulting from the Acc/Dec processor. Without accounting for friction, the application of the

Acc/Dec processor does not provide an accurate time representation of the actual feed rate. This issue needs to be investigated further.

Appendix E

Internet Based Implementation

E.1 Introduction

This appendix presents the concept and implementation of an Internet based facility for multi-axis milling process simulation and optimization. The implementation methodology is presented in section E.2. A detailed description of how to run the simulation is given in section E.3. Section E.4 describes how the Internet-based implementation of the multi-axis milling process simulation and optimization is included with other related machining technologies to provide a formalized system for designing and proofing manufacturing processes. Finally, a summary of the Internet based implementation is given in section E.5.

E.2 Implementation Methodology

The Internet implementation of the multi-axis milling simulation is based on a client/server model. The client, running on the user's web browser, makes requests for services. The server, running on the host computer, services the requests. The client and server are on different machines across the Internet. Communication between the client and server uses the TCP/IP (Internet Protocol).

The client-side (user interface) is implemented as a Java applet and some html pages. Java is a programming language similar to C++, but it is platform independent, allowing the same code to run on any computer that is Java-enabled. Applets are small Java programs that can be down-loaded to and displayed by a Java-enabled browser. As Java is a full featured programming language, it is easy to create truly interactive applications using it. Also, using an applet allows a lot of the work of setting up for the simulation and displaying results to be done on the client's computer. For the most part, the server's computer just has to accept input parameters, run the simulation and down-load results. Note that although the interpreted nature of Java allows it to be platform independent, there is a price that must be paid in terms of the speed at which it runs. As it is only being used for a user interface for this application, speed is not a concern.

The server-side is implemented using the Apache web server. It uses Java servlets, which are plug-in modules to the Apache web server to handle the applet's requests. As Java was created with the Internet in mind, it has useful classes that allow it to easily deal with communicating over the Internet, making programming of the applet to servlet communication easier. As the applet and servlet use browsers and Apache web server for their base, a great deal of the underlying work is done, such as the actual transmission of the data, through a TCP/IP connection and the display and handling of html pages and Java applets. The implementation work required to develop the Internet implementation of the multi-axis milling simulation is then restricted to just the milling simulation specific code.

The actual multi-axis milling simulation was written in C++, to take advantage of the speed of a compiled language. As it resides on the server computer, it does not need to be platform independent. Its input is a text setup file, and CLData files (tool path files). Its output is files of raw data points (x-y). In order to make the simulation easier to use, the applet provides a graphical user interface for specifying the setup file parameters, for selecting the CLData files to use and for graphically displaying the results. The servlets get the simulation inputs from the applet, formats them for the simulation, and launches and monitors the simulation.

There are several steps the client must take in order to run a simulation. These steps require the applet and servlets to communicate over the Internet a number of times. Additionally, the servlets can be handling several applets at a time. It must, therefore, know which applet is making a request. It does this by assigning a session number (key) to each applet upon request. All subsequent requests from the applet supply the session number it was given. This allows tracking of a given computer across multiple interactions. The applet allows the client to see and enter the key. This was desired because the simulations can take a long time to run. It is possible that the client would start a simulation and several hours later check its progress or retrieve the results at a different computer, such as from home. All they need to do this is the key.

E.3 Running the Simulation

The use of the simulation was kept as straight forward as possible. It requires the following steps:

1. Launch the applet: Start up the applet by going to its Uniform Resource Locator (URL), in your web browser. This down-loads the applet to the client's browser, as shown in Figure E.1.
2. Create a session: Request a session number. This causes the servlet to create a directory for this session number and copy required files to it, as shown in Figure E.2. This directory is used to store all data related to the given session, including simulation setup file, CLData files and results files.
3. Select Parameters: The client must specify the simulation parameters from the input screens, shown in Figure E.3, and upload the CLData files that the simulation is to use (or use the default demo.cl). Uploading files is done using an html page. This is done because the applet is untrusted and cannot directly access the client's hard drive due to security issues.
4. Run the Simulation: Once the user has specified all the required parameters, they can start a simulation run. This uploads the parameter list to the servlet. The setup file used by the simulation is created from the parameter list and the simulation is started.
5. Monitor the Simulation: The user can find out if the simulation is done or running from the status window. They can also abort a simulation in progress.
6. View the Results: The results can be viewed on the results screen, shown in Figure E.4, as graphs, raw data or computed data.

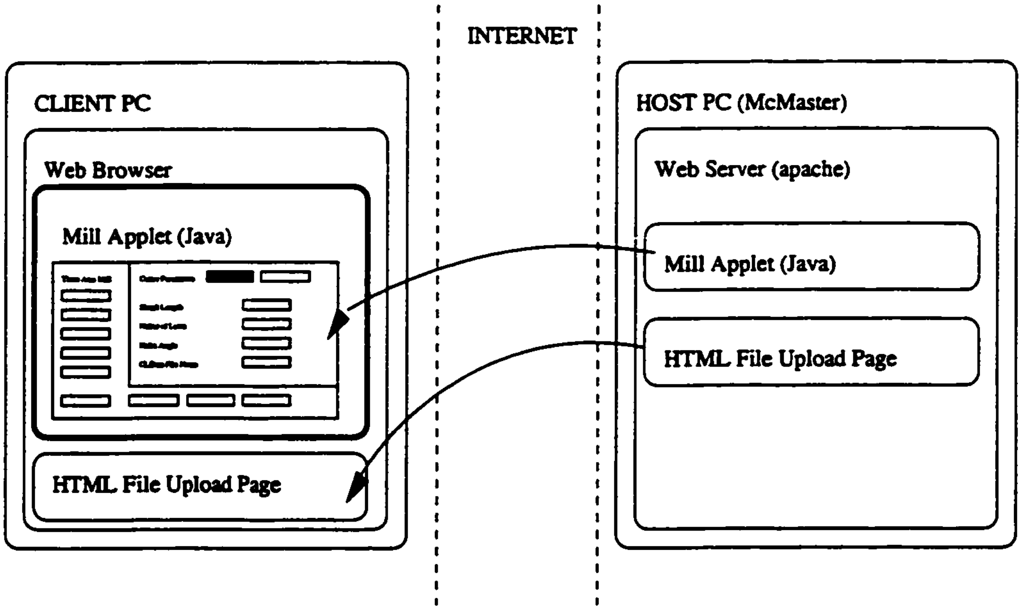


Figure E.1: Launching Applet

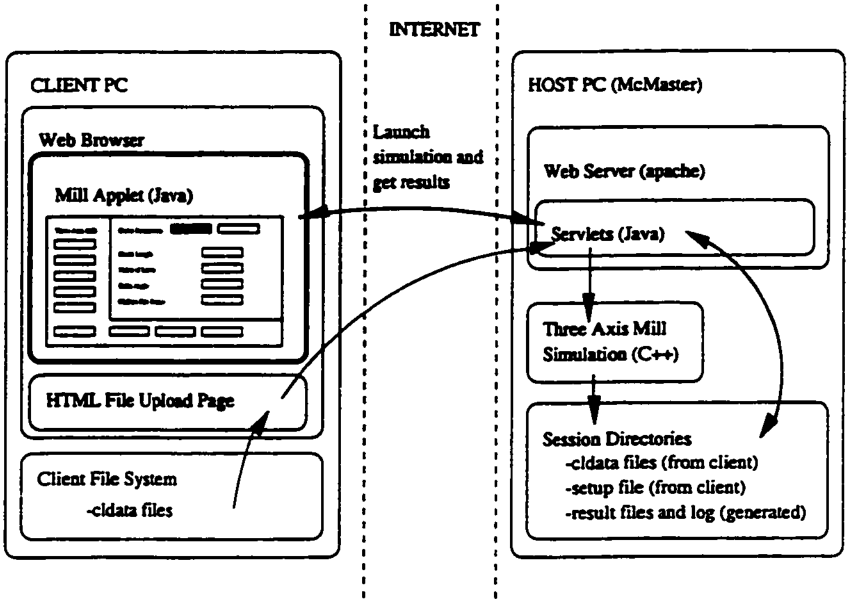
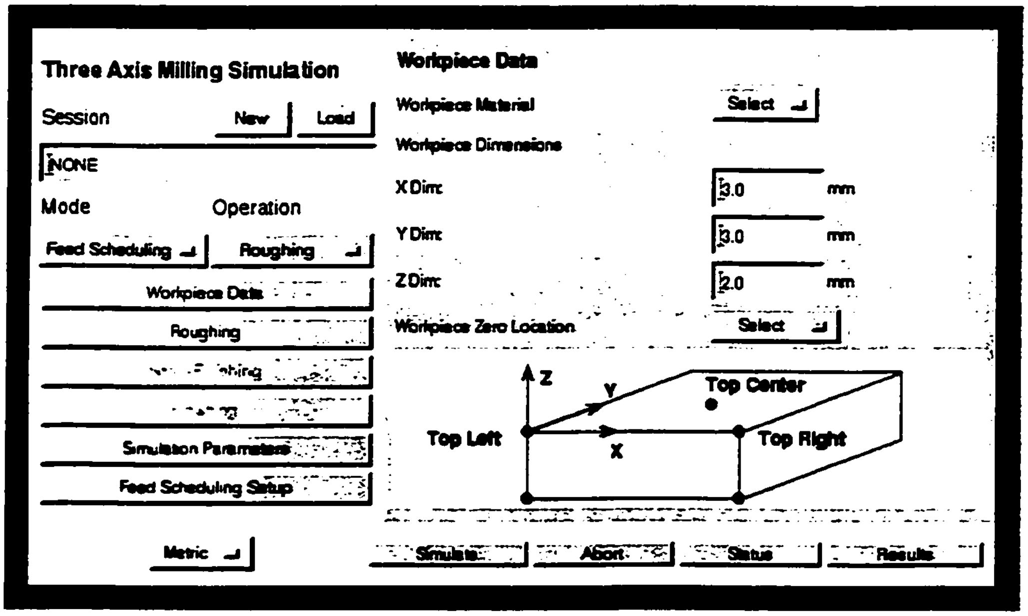


Figure E.2: Creating a Session



Session: NONE

Mode: Feed Scheduling

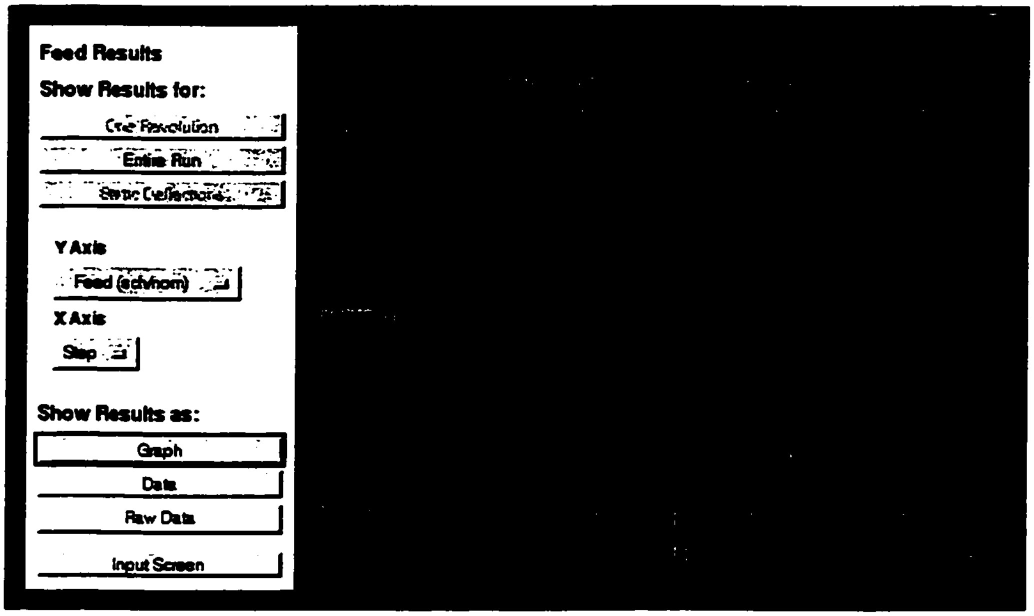
Operation: Roughing

X Dim: 3.0 mm

Y Dim: 3.0 mm

Z Dim: 2.0 mm

Figure E.3: Input Parameter Screen



Y Axis: Feed (ech/nom)

X Axis: Step

Figure E.4: Results Screen

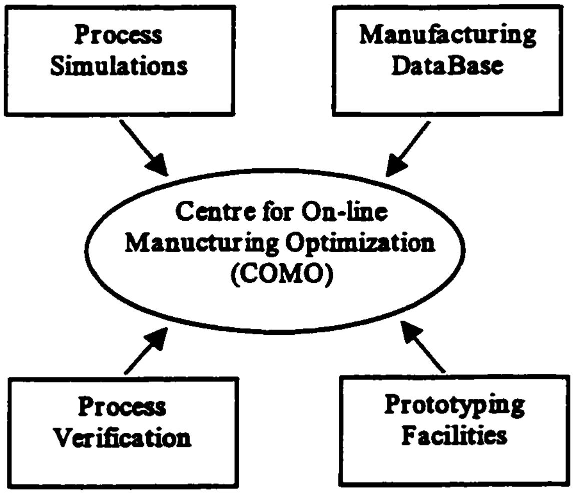


Figure E.5: Manufacturing Process Development Tools

E.4 Centre for On-line Machining Optimization (COMO)

The Internet-based implementation of the multi-axis milling process simulation and optimization is included with other related machining technologies to establish a Centre for On-line Manufacturing Optimization (COMO) at the McMaster Manufacturing Research Institute (MMRI). The primary focus of COMO is to provide the manufacturing engineer with a formalized system for designing and proofing manufacturing processes. The Centre was established in response to the lack of available tools for manufacturing process development. There are several key machining technologies that are included within COMO; manufacturing databases, process simulations, process verification, and prototyping services. These manufacturing process development tools are depicted in figure E.5.

The manufacturing database is used to archive tooling information (types, material properties, geometry, etc.), workpiece specification data (material properties, machinability ratings, etc.), calibration data (cutting force coefficients, tool static and dynamic deflection constants, etc.), and manufacturing process history. The manufacturing data base can be accessed at different levels. The manufacturing engineer accesses the database directly to get information related to tooling, workpiece and manufacturing process history. The process simulations access the database to obtain tooling and workpiece specific information as well as calibration data. The final level of access allows database modifications. Access at this level is restricted to the COMO administrator.

The process simulations that are available within COMO include; multi-axis milling, turning, drilling, and tapping. The multi-axis milling simulations inlude ball and flat peripheral milling and face milling. The turning simulation includes 2-axis contouring capabilities. The drilling simulations include twist and spade drills. The tapping simulations include straight fluted and spiral fluted taps.

The process verification module is used to anaylyze part fixtures, evaluate tooling designs, and to ensure that manufacturing tolerances are attainable.

The prototyping facility at COMO provides access to a number of machine tools ranging from conventional CNC lathes and mills to ultra-high-speed machining centres. The machine tools are instrumented with force, vibration, temperature, and dimensional measuring devices. An Open Architecture Machine (OAC) controller has been implemented on both a turning centre and machining centre. The OAC provides an open platform for integrating advanced process monitoring and control applications. Part quality is assessed through various guaging systems including a

CMM. Both prototype parts or small batch production runs can be accomodated by COMO.

COMO provides the manufacturing engineer with a formalized system for designing and proofing manufacturing processes. Process develoment can occur iteratively, without the need for lengthy and/or expensive process trials. Implementing these development tools using Internet technologies has several advantages. It allows all of the development tools to be in a centralized location, allowing for quick and reliable upgrade paths. Internet accessibility also gives quick and easy access to these development tools to many manufacturing engineers around the world. COMO is intended for manufacturing process development and as a virtual training platform for tomorrow's manufacturing engineer.

E.5 Summary

This appendix presented the concept and implementation of an Internet-based facility for multi-axis milling process simulation and optimization. The facility integrates advanced process simulation software, and state of the art multimedia and Internet technologies to realize an Internet enabled process optimization environment for machining systems. This appendix also described how the Internet-based implementation of the multi-axis milling process simulation and optimization is included with other related machining technologies to provide a formalized system for designing and proofing manufacturing processes.

References

[1] F. Abrari and M.A. Elbestawi. Closed Form Formulation of Cutting Forces for Ball and Flat End Mills. *Int. Journal of Machine Tool Design and Manufacture*, 37(1):17–27, 1996.

[2] Farid Abrari. A Regenerative Dynamic Force Model for Ball End Milling. Master's thesis, McMaster University, 1994.

[3] Farid Abrari. *MultiAxis Milling of Flexible Parts.* PhD thesis, McMaster University, 1998.

[4] Y. Altintas and J. Peng. Design and Analysis of a Modular CNC System. *Computers in Industry*, 13:305–316, 1990.

[5] R.O. Anderson. Detecting and Eliminating Collisions in ND Machining. *Computer Aided Design*, 10(4):231–237, 1987.

[6] R. Ardekani and I. Yellowley. The Control of Multiple Constraints Within an Open Architecture Machine Tool Controller. *ASME Journal of Manufacturing Science and Engineering*, 118(8):388–393, 1996.

[7] E. Armarego and N. Deshpande. Computerized Predictive Cutting Models for Forces in End-Milling Including Eccentricity Effects. *Annals of CIRP*, 38(1):45–49, 1989.

[8] E. Armarego, J. Smith, and J. Wang. Computer Aided Contrained Optimization Analysis and Strategies for Multipass Helical Tooth Milling Operations. *Annals of CIRP*, 43(1):437–442, 1994.

[9] 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 Analysis. *Int. Journal of Production Research*, 31(8):1991–2009, 1993.

[10] 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. *Int. Journal of Production Research*, 31(10):2319–2336, 1993.

[11] K. Astrom and B. Wittenmark. *Computer Controlled systems.* Prentice-Hall, Englewood Cliffs, N.J., 1984.

[12] S. Bedi and E. Cohen. Toolpath Generation for Freeform Surface Models. *Computer Aided Design*, 18:307–313, 1994.

[13] J.G. Bollinger and N.A. Duffie. *Computer Control of Machines and Processes.* Addison-Wesley Publishing Company, 1989.

[14] G.E.P Box, W.G. Hunter, and J.S. Hunter. *Statistics for Experimenters.* John Wiley and Sons, New York, 1978.

[15] G.E.P. Box and G.M. Jenkins. *Time Series Analysis: Forecasting and Control.* Holden-Day, 1976.

[16] 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.

[17] J. Butler, B. Haak, and M. Tomizuka. Reference Input Generation for High Speed Coordinated Motion of a Two Axis System. *ASME Journal of Dynamic Systems, Measurment and Controls*, 113(1):67–74, 1991.

[18] R.M. Centner and J.M. Idelson. Adaptive Controller for a Metal Cutting Process. In *Joint Automatic Control Conference*, pages 154–161. IEEE/ASME, 1964.

[19] I.T. Chappel. The Use of Vectors to Simulate Material Removed by Numerically Controlled Milling. *Computer Aided Design*, 15(3):156–158, 1983.

[20] C.S. Chen and A.C. Lee. Design of Acceleration/Deceleration Profiles in Motion Control Based on Digital FIR Filters. *Int. Journal of Machine Tool Design and Manufacture*, 38:799–825, 1998.

[21] C.N. Chu, S.Y. Kim, and J.M. Lee. Feed-Rate Optimization of Ball End Milling Considering Local Shape Features. *Annals of CIRP*, 46(1):433–436, 1997.

[22] H.Y. Chuang and C.H. Liu. Cross-Coupled Adaptive Feedrate Control for Multi-Axis Machine Tools. *ASME Journal of Dynamic Systems, Measurment and Controls*, 113:451–457, September 1991.

[23] S. Chuang and W. Lin. Tool Path Generation for Pockets with Free Form Curve using Bezier Convex Hulls. *Int. Journal of Advanced Manufacturing Technology*, 13(2):109–115, 1997.

[24] L.K. Daneshmend and H.A. Pack. Model Reference Adaptive Control of Feed force in Turning. *ASME Journal of Dynamic Systems, Measurment and Controls*, 108:215–222, 1986.

[25] R.E. DeVor, S.G. Kapoor, E.C. DeMeter, and Q. Sayeed. An Internet-Based Model for Technology Integration and Access, Part 2: Applications to Process Modeling and Fixture Design. *ASME Journal of Manufacturing Science and Engineering*, 2(2):1005–1022, 1995.

[26] R.E. DeVor, S.G. Kapoor, and G. Venkatasubramanian. An Internet-Based Model for Technology Integration and Access, Part 1: Software and Hardware Testbed Models. *ASME Journal of Manufacturing Science and Engineering*, 2(2):991–1004, 1995.

[27] K.F. Ehmann, W.S. Yun, and D.W. Cho. Determination of Constant 3D Cutting Force Coefficients and of Runout Parameters in End Milling. *Transactions of NAMRI/SME*, 27:87–92, 1999.

[28] H. El-Mounayri, A.D. Spence, and M.A. Elbestawi. Enhanced CAD/CAM for Simulation and Optimization of 3-5 Axis Millng of Dies and Molds. In *Proc. Conf. Of CSME*, McMaster University, 1996. CSME.

[29] H. El-Mounayri, A.D. Spence, and M.A. Elbestawi. Milling Process Simulation - A Generic Solid Modeller Based Paradigm. *ASME Journal of Manufacturing Science and Engineering*, 120:213–221, 1998.

[30] G. Elbert and E. Cohen. Toolpath Generation for Freeform Surface Models. *Computer Aided Design*, 26(6):490–496, 1994.

[31] M.A. Elbestawi, L. Liu, and N.K. Sinha. Some Advanced Control Strategies for Modern Machine Tools. *Computers in Industry*, 16:47–57, 1991.

[32] H.Y. Feng and C. Meng. The Prediction of Cutting Forces in the Ball End Milling Process, Part 1: Model Formulation and Model Building Procedure. *Int. Journal of Machine Tool Design and Manufacture*, 34(5):697–710, 1993.

[33] G. Frain. *Curves and Surfaces for CAGD.* Academic press, San Diego,CA, first edition, 1992.

[34] B.K. Fussel, J.G. Hemmett, and R.B. Jerard. Modeling of Five-Axis End Mill Cutting Using Axially Discretized Tool Moves. *Transactions of NAMRI/SME*, 27:81–86. 1999.

[35] Y. Haung and J.H. Oliver. Non-Constant Parameter NC Tool Path Generation On Sculptured Surfaces. *Int. Journal of Advanced Manufacturing Technology*, 9:281–290. 1994.

[36] 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):189–202, 1994.

[37] T. Hosoi. Cutting Action of Ball End Mill with a Spiral Edge. *Annals of CIRP*, 25(1):49–53, 1977.

[38] S.J. Huang and C.C. Chen. Application of Self-Tuning Feed-Forward and Cross-Coupling Control in a Retrofitted Milling Machine. *Int. Journal of Machine Tool Design and Manufacture*, 35(4):577–591, 1995.

[39] Behnam Imani. *Model Based Die Cavity Machining Simulation Methodology.* PhD thesis, McMaster University, 1998.

[40] F. Ismail and J. Tlusty. Special Aspects of Chatter in Milling. *ASME Journal of Vibration, Acoustics, Stress, and Reliability in Design*, 105(1):24–32, 1983.

[41] S. Jain and D.C. Yang. A Systematic Force Analysis of the Milling Operation. In *ASME Winter Annual Meeting*, pages 55–63. ASME, 1989.

[42] R.B. Jerrard, R.L. Drysdale, and K. Haulk. Geometric Simulation for Numerical Control Machining. In *Proc. ASME Int'l Computer in Engineering Conference*, volume 2, pages 129–136, San Francisco, Jul 31-Aug 3 1988. ASME.

[43] S.G. Kapoor, R.E. DeVor, R. Zhu, R. Gajjela, G. Parakkal, and D. Smithey. Development of Mechanistic Models for the Prediction of Machining Performance:

Model-Building Methodology. In *CIRP International Workshop on Modeling of Machining Operations*, number 2 in 2A, pages 1–12, May 1998.

[44] K. Kim and Jeong. Tool Path Generation for Machining Free Form Pockets with Islands. *Computers and Industrial Engineering*, 28(2):399–407, 1995.

[45] K.F. Koch, B. Lilly, E. Kropp, and T. Altan. Development of a CAE-Module for Calculating Cutting Forces in 3-Axis Milling of Sculptured Surfaces in Die Manufacturing. Technical Report ERC/NSM-D-90-43, Engineering Research Center for Net Shape Manufacturing, Ohio State University, 1990.

[46] Y. Koren. Cross-Coupled Biaxial Computer Controls for Manufacturing Systems. *ASME Journal of Dynamic Systems, Measurment and Controls*, 102(1):265–272, 1980.

[47] Y. Koren. Control of Machine Tools. *ASME Journal of Manufacturing Science and Engineering*, 119:749–755, November 1997.

[48] Y. Koren and C.C. Lo. Variable-Gain Cross-Coupling Controller for Contouring. *Annals of CIRP*, 40(1):371–374, 1991.

[49] Y. Koren and C.C. Lo. Advanced Controllers for Feed Drives. *Annals of CIRP*, 41(1):689–698, 1992.

[50] P.K. Kulkarni and K. Srinivasan. Identification of Discrete Time Dynamic Models for Machine Tool Feed Drives. In *Symposium on Sensors and Controls for Automated Manufacturing and Robotics*, pages 1–12, New Orleans, LA., 1984. ASME Winter Annual Meeting.

[51] P.K. Kulkarni and K. Srinivasan. Optimal Contouring Control of Multi-Axial Feed Drive Servomechanisms. *ASME Journal of Engineering for Industry*, 111(2):140–149, 1989.

[52] P.K. Kulkarni and K. Srinivasan. Cross-Coupled Control of Biaxial Feed Drive Servomechanisms. *ASME Journal of Dynamic Systems, Measurment and Controls*, 112:225–232, June 1990.

[53] L.K Lauderbaugh and A.G. Ulsoy. Dynamic Modeling for Control of the Milling Process. In *Sensors and Controls for Manufacturing*, volume 18, pages 149–158. ASME Winter Annual Meeting, 1985.

[54] I. Lazoglu and S.Y. Liang. An Improved Analytical Modeling of Force System in Ball-End Milling. In *Proc. Conf. Of CSME*, McMaster University, 1996. CSME.

[55] Y.S. Lee. Non-isoparametric Tool Path Planning by Machining Strip Evaluation for 5-Axis Sculptured Surface Machining. *Computer Aided Design*, 30(7):559–570, 1998.

[56] S.X. Li and R.B. Jerard. 5-Axis Machining of Sculptured Surfaces with Flat-end cutter. *Computer Aided Design*, 26(3):165–178, 1994.

[57] E.M. Lim, H.Y. Feng, and C. Meng. The Prediction of Dimensional Errors for Machining Sculptured Surfaces using Ball-End Milling. *ASME Journal of Manufacturing Science and Engineering*, 64:149–456, 1993.

[58] E.M. Lim and C.H. Menq. Integrated Planning For Precision Machining of Complex Surfaces Part1: Cutting-Path and Feedrate optimization. *Int. Journal of Machine Tool Design and Manufacture*, 37(1):61–75, 1997.

[59] L. Ljung. *System Identification : Theory for the User.* Prentice-Hall, Englewood Cliffs, N.J., 1987.

[60] H.Y. Maeng, M.H. Ly, and G.W. Vickers. Feature-Based Machining of Curved Surfaces Using the Steepest Directed Tree Approach. *Journal of Manufacturing Systems*, 15(6):379–391, 1996.

[61] S. Marshall and J.G. Griffiths. A Survey of Cutter Path Construction Techniques for Milling Machines. *Int. Journal of Production Research*, 32(12):2861–2877, 1994.

[62] M.E. Martelloti. An Analysis of the Milling Process. *Transactions of ASME*, 63:677–700, 1941.

[63] MATLAB. *MATLAB User's Manual.* The Math Works Inc, 1993.

[64] Metcut. *Machining Data Handbook*, volume 1. Metcut, 3 edition, 1980.

[65] MTAMRI. Softwaretestbeds. http://mtamri.me.uiuc.edu/testbeds, 2000.

[66] D.E. Newland. *An Introduction to Random Vibrations.* John Wiley and Son, England, Longman Scientific&Technical.

[67] S.M. Pandit. Stochastic Linearization by Data Dependent Systems. *ASME Journal of Dynamic Systems, Measurment and Controls*, 99:221–226, December 1977.

[68] A. Pankratz. *Forecasting with Dynamic Regression Models*. John Wiley and Son, 1991.

[69] L. Piegl and W. Tiller. *The NURBS Book*. Springer, New York, second edition, 1997.

[70] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, MA, second edition, 1999.

[71] G. Pritschow and W. Philipp. Research on the efficiency of the feedforward controllers in M direct drives. *Annals of CIRP*, 41(1):411–415, 1992.

[72] N. Rao, S. Bedi, and R. Buchal. Implementation of the Principal-Axis Method for Machining of Complex Surfaces. *Int. Journal of Advanced Manufacturing Technology*, 11:249–257, 1996.

[73] S.J. Robert and Y.C. Shin. Control of Cutting Force for End Milling Processes Using an Extended Model Reference Adaptive Control Scheme. *ASME Journal of Manufacturing Science and Engineering*, 118(8):339–347, 1996.

[74] A. Sabberwal. Chip Section and Cutting Force During the Milling Operation. *Annals of CIRP*, 18(1):197–203, 1961.

[75] C.S Smith and P.K. Wright. CyberCut: A World Wide Web Based Design-to-Fabrication Tool. *Journal of Manufacturing Systems*, 15(6):432–442, 1996.

[76] S. Smith and J. Tlusty. An Overview of Modeling and Simulation of the Milling Process. *ASME Journal of Engineering for Industry*, 113(1):169–175, 1991.

[77] S. Smith and J. Tlusty. An Overview of Modeling and Simulation of the Milling Process. *ASME Journal of Engineering for Industry*, 113(1):169–175, 1991.

[78] A. Spence and Y. Altintas. End Milling Force Algorithm for CAD Systems. *Annals of CIRP*, 41(1):31–34, 1991.

[79] A.D. Spence and Y. Altintas. A solid Modeler Based Millling Process Simulation and Planning System. *ASME Journal of Engineering for Industry*, 116:61–69, 1994.

[80] K. Srinivasan and T.C. Tsao. Machine Tool Feed Drives and Their Control - A Survey of the State of the Art. *ASME Journal of Manufacturing Science and Engineering*, 119:743–748, September 1997.

[81] J.W. Sutherland. An Improved Method for Cutting Force and Surface Error Prediction in Flexible End Milling Systems. *ASME Journal of Engineering for Industry*, 108:269–279, 1986.

[82] J.W. Sutherland, T. Cao, C.M. Daniel, Y. Yue, and Y. Zheng. An Internet-Based Cutting Fluid Evaluation Software Testbed. *Transactions of NAMRI/SME*, 15:243–248, 1997.

[83] S. Takata. Generation of a Machining Scenario and Its Applications to Intelligent Machining Operations. *Annals of CIRP*, 42(1):531–534, 1993.

[84] S. Takata, M.D. Tsai, and T. Sata. A Cutting Simulation System for Machinability, Evaluation Using a Workpiece Model. *Annals of CIRP*, 38(1):417–420, 1989.

[85] M. Tomizuka. Zero Phase Error Tracking Algorithm for Digital Control. *ASME Journal of Dynamic Systems, Measurment and Controls*, 109(1):65–68, 1987.

[86] M. Tomizuka. On the Design of Digital Tracking Controllers. *ASME Journal of Dynamic Systems, Measurment and Controls*, 115:412–418, June 1993.

[87] M. Tomizuka, J.H. Oh, and D.A. Dornfeld. Model Reference Adaptive Control of the Milling Process. In *Control of Manufacturing Processes and Robotics Systems*, pages 55–63. ASME Winter Annual Meeting, 1983.

[88] T.C. Tsao and M. Tomizuka. Adaptive Zero Phase Error Tracking Algorithm for Digital Control. *ASME Journal of Dynamic Systems, Measurment and Controls*, 109(4):349–354, 1987.

[89] E.D. Tung, G. Anwar, and M. Tomizuka. Low Velocity Friction Compensation and Feedforward Solution Based on Repetitive Control. *ASME Journal of Dynamic Systems, Measurment and Controls*, 115:279–284, June 1993.

[90] E.D. Tung and M. Tomizuka. Feedforward Tracking Controller Design Based on the Identification of Low Frequency Dynamics. *ASME Journal of Dynamic Systems, Measurment and Controls*, 115:348–356, September 1993.

[91] E.D. Tung, M. Tomizuka, and Y. Urishisaki. High-Speed End Milling Using a Feedforward Control Architecture. *ASME Journal of Manufacturing Science and Engineering*, 118:178–187, May 1996.

[92] H.B. Voelcker and W.A. Hunt. The Role of Solid Modeling in Machining-Process Modeling and NC Verification. In *Int'l Congress and Exposition*, pages 1–8. Society of Automotive Engineers, Feb 1981.

[93] J.J. Wang and S.Y. Liang. Convolution Analysis of Milling Force Pulsation. *ASME Journal of Engineering for Industry*, 116(1):17–25, 1994.

[94] J.J. Wang and S.Y. Liang. Chip Load Kinematics in Milling with Radial Cutter Runout. *ASME Journal of Engineering for Industry*, 118(1):111–116, 1996.

[95] W.P. Wang. Solid Modeling for Optimizing Metal Removal of Three Dimensional End-Milling. *Journal of Manufacturing Systems*, 7(1):57–65, 1988.

[96] W.P. Wang and K.K. Wang. Geometric Modeling for Swept Volume of Moving Solids. *IEEE Computer Graphics and Applications*, 12:8–17, 1986.

[97] T. Watanabe. A Model-Based Approach to Adaptive Control Optimization in Milling. *ASME Journal of Dynamic Systems, Measurment and Controls*, 108(3):56–64, 1986.

[98] M. Weck and G. Ye. Sharp Corner Tracking Using the IKF Control Strategy. *Annals of CIRP*, 39(1):437–441, 1990.

[99] K. Weinert, A. Enselmann, and J. Friedhoff. Milling Simulation for Process Optimization in the Field of Die and Mould Manufacturing. *Annals of CIRP*, 46(1):325–328, 1997.

[100] P.D. Welch. *The Use of Fast Fourier Transform for the Estimation of Power Spectra : A Method Based on Time Averaging Over Short, Modified Periodograms, Modern Spectrum Analysis*, pages 17–20. John Wiley and Son, New York, 1978.

[101] P.K. Wright and D. Dornfeld. CyberCut: A Networked Machining Service. *Transactions of NAMRI/SME*, 16:281–286, 1998.

[102] M. Yamada, Y. Funahashi, and Z. Riadh. Generalized Optimal Zero Phase Error Tracking Controller Design. *ASME Journal of Dynamic Systems, Measurment and Controls*, 121:165–170, June 1999.

[103] M. Yang and H. Park. The Prediction of Cutting Forces in Ball-End Milling. *Int. Journal of Machine Tool Design and Manufacture*, 31(1):45–54, 1991.

[104] Z. Yazar. K.F. Koch, T. Merrick, and T. Altan. Feed Rate Optimization Based on Cutting Force Calculions in 3-Axis Milling of Dies and Molds with Sculptured Surfaces. *Int. Journal of Machine Tool Design and Manufacture*, 34(3):365–377, 1994.

[105] G. Yucesan and Y. Altintas. Mechanics of Ball End Milling Process. *ASME Journal of Manufacturing Science and Engineering*, 64:543–551, 1993.

[106] G. Yucesan and Y. Altintas. Prediction of Ball End Milling Forces. *ASME Journal of Engineering for Industry*, 118:95–103, 1996.

[107] G. Yucesan, A.E. Bayoumi, and L.A. Kendall. An Analytic Cutting Force Model for Milling. *Transactions of NAMRI/SME*, pages 137–145, 1990.

[108] L. Zheng and S.Y. Liang. Analysis of end Milling Forces with Cutter Axis Tilt. *Transactions of NAMRI/SME*, 23:137–142, 1995.

[109] L. Zheng and S.Y. Liang. Identification of Cutter Axis Tilt in End Milling. *ASME Journal of Engineering for Industry*, 119(1):178–185, 1997.