Prediction and Reduction of Cycle Time for Five-Axis CNC Machine Tools

by

Sneha Tulsyan

B.Tech.(Hons.), Indian Institute of Technology Kharagpur, India, 2011

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

Master of Applied Science

in

THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

November 2014

© Sneha Tulsyan, 2014

Abstract

The growing demand for high speed machining in aerospace, automotive and die and mold industry has directed the interest of research community towards prediction and reduction of machining cycle time. In this thesis, a cycle time prediction scheme is proposed for milling operations based on identified CNC machine dynamics in exact-stop and continuous mode. Various system identification techniques are utilized to identify the implemented trajectory generation and corner smoothing technique and feed drive dynamics of the CNC system. An analytical approach for predicting cycle time based on the identified CNC system dynamics and given part program is presented. It is shown that the cycle time of NC machining process is predominantly affected by trajectory generation and corner smoothing techniques does not have much influence on the cycle time, since the tracking delay is insignificant in position control servos. The proposed algorithm is validated in experiments and experimental results has shown that the cycle time prediction error remains within 5% for various 2-axis, 3-axis and 5-axis toolpaths.

In the later half of the thesis, a new decoupled approach for five-axis corner smoothing is presented to reduce the cycle time of milling operations. Toolpath position and orientation are smoothed by inserting quintic and normalized septic micro-splines, respectively between the adjacent linear toolpath segments. Optimal control points are calculated for position and orientation splines to achieve C^3 continuity at the junctions between the splines and the linear segments while respecting user-defined corner position tolerance and orientation tolerance limits. Synchronization of position and orientation splines is carried out. After geometrical modification of the toolpath, feedrate planning is performed using C^3 continuous cubic acceleration feedrate profile to preserve jerk continuity in toolpath motion. The proposed C^3 continuous toolpath motion is compared against the unsmooth and C^2 continuous motion in experiments and simulations to show improvements in cycle time, tracking accuracy and smoothness throughout the toolpath.

Preface

This dissertation is original intellectual property of the author, Sneha Tulsyan, under supervision of Professor Yusuf Altintas. This work has been completed in the Manufacturing Automation Laboratory at the University of British Columbia.

A version of Chapter 3 is presented in Machine Tool Technologies Research Foundation (MTTRF) IAM-CNC Annual Meeting 2014. The work is also published in the proceedings of MTTRF 2014 meeting [Tulsyan S, Yuen A, Altintas Y. Estimation of NC Cycle Time for Free-Form Milling Operations on Commercial CNC Machining Centers. The Proceedings of MTTRF 2014 Meeting, pp.239-244]. I was the lead investigator, responsible for experiment design, data collection and the major development of identification algorithm and manuscript. Alexander Yuen was responsible for editing the manuscript. Professor Yusuf Altintas was the supervisory author on this project and was involved in problem statement formulation, supervision throughout the project and manuscript editing.

A version of Chapter 4 is going to be submitted for publication.

Table of contents

Abstractii
Prefaceiii
Table of contents iv
List of tablesvii
List of figures viii
Acknowledgements xi
Chapter 1: Introduction1
Chapter 2: Literature review
2.1 Overview
2.2 Cycle time prediction for CNC milling operation
2.2.1 Using NC program information
2.2.2 Using material removal rate
2.2.3 Using machine characteristics and part program
2.3 Geometrical toolpath smoothing
2.3.1 Global toolpath smoothing
2.3.2 Local toolpath smoothing 10
2.3.2.1 Three-axis corner smoothing
2.3.2.2 Five-axis corner smoothing
2.4 Summary 12
Chapter 3: Cycle time prediction for milling process14

3.1	Overview1	4
3.2	Modeling dynamics of CNC systems 1	5
3.2.	.1 Identification of trajectory parameters	8
3.2.2	.2 Prediction of corner smoothing technique	27
3.2.	.3 Identification of closed-loop feed-drive dynamics	28
3.3	Cycle time estimation	36
3.4	Experimental validation ²	¥1
3.5	Summary ²	45
Chapter	e 4: Five-axis corner smoothing for CNC system	1 7
4.1	Overview ²	1 7
4.2	Toolpath position smoothing ⁴	18
4.2.	.1 Parametric position micro-spline ⁴	18
4.2.2	.2 Optimal control point calculation for position micro-spline	53
4.	.2.2.1 Acceleration and jerk continuity at junctions	54
4.	2.2.2 Pre-defined position error tolerance limit, εpos	59
4.3	Toolpath orientation smoothing ϵ	52
4.3.	.1 Parametric orientation micro-spline ϵ	54
4.3.2	.2 Optimal control point calculation for orientation micro-spline \dots	56
4.	.3.2.1 Velocity, acceleration and jerk continuity at junctions	57
4.	.3.2.2 Pre-defined orientation error tolerance limit, ɛori	14
4.	.3.2.3 Synchronisation of position and orientation spline	15
4.4	Feedrate planning	78

4.5	Experimental results	
4.6	Summary	
Chapte	r 5: Conclusions and future work	95
5.1	Conclusions	
5.2	Future research directions	
Bibliog	raphy	

List of tables

Table 3.1 Trust region parameter estimates	24
Table 3.2 Identified closed loop feed-drive dynamics of Mori Seiki NMV5000 CNC system	34
Table 3.3 Experimental and predicted cycle time	45
Table 4.1 Plant parameters for Fadal VMC 2216	86
Table 4.2 PID controller parameters	86
Table 4.3 Percentage improvement in tracking error	93

List of figures

Figure 3.1 Flow of commands in Commercial CNC System 1	5
Figure 3.2 Feedrate profile for exact-stop and continuous mode 1	6
Figure 3.3 Configuration of Mori Seiki NMV5000 DCG commercial machining center [29] 1	7
Figure 3.4 Kinematic profiles of Mori Seiki NMV5000 1	9
Figure 3.5 Convergence of cost function in Trust-region algorithm	3
Figure 3.6 Convergence of Trust-region size in Trust-region algorithm	3
Figure 3.7 Experimental feedrate profile in (i) Exact-stop mode, (ii) Continuous mode 2	7
Figure 3.8 Predicted feedrate profile in Continuous mode	8
Figure 3.9 Control block diagram of Mori Seiki NMV5000 feed-drive system	9
Figure 3.10 Experimental velocity loop FRF of X-drive of Mori Seiki NMV5000 DCG CNC	
	0
system	0
Figure 3.11 Experimental and identified closed loop frequency response function of X-axis 3	3
Figure 3.11 Experimental and identified closed loop frequency response function of X-axis 3 Figure 3.12 Experimental and identified closed loop FRF	3
 System	3 5 0
system 3 Figure 3.11 Experimental and identified closed loop frequency response function of X-axis	5 .0
system 3 Figure 3.11 Experimental and identified closed loop frequency response function of X-axis	.0 .0 .1
system 3 Figure 3.11 Experimental and identified closed loop frequency response function of X-axis	.0 .0 .1 .2
system 3 Figure 3.11 Experimental and identified closed loop frequency response function of X-axis	.0 .0 .1 .3 .3
system 3 Figure 3.11 Experimental and identified closed loop frequency response function of X-axis	.0 .0 .1 .2 .3 .4

Figure 4.2 Decoupled method of five-axis corner smoothing
Figure 4.3 Plot of basis function corresponding to different control points for quintic B-Spline
Figure 4.4 Toolpath position smoothing using quintic B-spline
Figure 4.5 Constraints for acceleration and jerk continuities at junctions
Figure 4.6 Location of position control points based on continuity and error tolerance constraints
Figure 4.7 Toolpath orientation smoothing using Normalized Heptic B-Spline
Figure 4.8 Basis functions of Heptic and Normalized Heptic B-Spline
Figure 4.9 Constraints for velocity, acceleration and jerk continuities at junctions for orientation
spline
Figure 4.10 Synchronization of position and orientation spline
Figure 4.11 Lengths of toolpath segments
Figure 4.12 Cubic acceleration feedrate profile
Figure 4.13 Different methods of interpolating along a given spline toolpath
Figure 4.14 Adaptive approach of formulating feed correction spline u(s)
Figure 4.15 Kinematic configuration of 5-axis CNC machining center
Figure 4.16 Experimental test bed consisting of the Fadal VMC 2216, Nikken 130-AX compact
tilting table and dSPACE open real-time controller
Figure 4.17 Control law and modelled plant for five-axis corner smoothing experiments
Figure 4.18 Test toolpath for five-axis corner smoothing experiments
Figure 4.19 Toolpath position and orientation error in reference command generation

Figure 4.20 Experimental tangential feedrate profile before and after corner smoothing	90
Figure 4.21 Jerk of axis reference commands for experiments	91
Figure 4.22 Experimental average axis tracking error from encoder measurements	92

Acknowledgements

I would like to express my sincere gratitude towards my supervisor Prof. Yusuf Altintas for the valuable guidance and support he has provided throughout my research and course work. As this is my first research experience, I would also like to thank him for enlarging my vision and mentoring me in the process of becoming an excellent researcher.

I would also like to thank my colleagues in Manufacturing Automation Laboratory for sharing their knowledge and experience with me, especially in the initial few months when I was not very familiar with the lab equipment. I would particularly like to thank Gurbachan S Sekhon and Alexander Yuen to help me carrying out experiments.

Finally, I would like to thank my family and friends for their endless moral support during my abroad study. I would like to dedicate this thesis to my beloved parents.

This research was sponsored by Natural Science and Engineering Research Council of Canadian Network for Research and Innovation in Machining Technology (NSERC CANRIMT) and Machine Tool Technologies Research Foundation (MTTRF).

Chapter 1: Introduction

Five-axis computer numerical control (CNC) machine tools are widely used for manufacturing of sculptured surfaces in aerospace, die and mold industries. It facilitates machining of complex surfaces which are otherwise difficult to machine using conventional three-axis CNC machine tools. In the context of five-axis CNC machining, there is a growing demand for the increased productivity as the manufacturing cost of the final product is directly proportional to the time taken to manufacture it. The prediction and optimization of cycle time for milling free form surfaces is therefore essential in comparing the performances of various five-axis machine tools.

Cycle time which essentially is the sum of primary cutting and auxiliary non-cutting motions for manufacturing a specific part, depends upon a number of factors such as toolpath, commanded feedrate, feedrate planning, geometry modification algorithm, CNC machine capabilities in terms of maximum achievable feedrate, acceleration, jerk, etc. Knowing the exact cycle time in advance leads to more accurate production process planning which in turn, greatly affects the performance of a shop floor from production economics perspective. Besides, it can also be used by product designers to improve and optimize the design of products and to find optimal tool path strategy to machine a component.

Parallel to cycle time prediction, research interest in the field of cycle time optimization has also gained momentum. Reducing the manufacturing cycle time has benefits including reduced cost, lower inventory, improved flexibility and faster response to customer order. One of the ways to address this is to improve the feedrate, acceleration and jerk limits of the drive. Other possibility is to optimize the toolpath. For a given machine tool and toolpath, the cycle time can be further improved by geometrical smoothing of the path which is otherwise, generally composed of a series of linear toolpath segments, also called as G01 commands. Traditionally, tool motion is allowed to cease at all the block transitions which results in higher cycle time and productivity suffers. It should be noted that if the tool is commanded to maintain its speed across the corners,

discontinuities are created in velocity, acceleration and jerk commands due to geometrical discontinuities in the toolpath which will result in high impact load on the feed drives causing machine tool vibrations and poor surface quality. In order to maintain a non-zero feedrate across corners while respecting feed drive acceleration and jerk bounds, the toolpath geometry needs to be smoothed such that the modified toolpath has finite curvature over its entire extent. Commercial CNC systems appear to have some corner smoothing algorithms such as injecting circular path at sharp corners. However, the smoothing needs to ensure that velocity, acceleration and jerk limits of the servo drives must not be violated. The objective of this research work is to develop models to predict and optimize machining cycle time of five-axis CNC machine tools by considering smooth trajectory transitions at sharp corners.

The rest of the thesis is organized as follows: The review of relevant literature in the field of cycle time prediction and toolpath smoothing is presented in chapter 2. In chapter 3, a new cycle time prediction algorithm based on NC program, trajectory profile, drive limits, geometry modification algorithm and machine characteristics is proposed for accurate estimation of machining time for milling free form surfaces. This is followed by a novel decoupled approach to achieve local corner smoothing by inserting micro-splines between adjacent linear toolpath segments for five-axis CNC machine tools. Constraint equations are formulated to ensure C^3 continuity (continuous jerk) throughout the toolpath while respecting user-defined position and orientation corner tolerance as presented in chapter 4. Chapter 5 concludes the thesis stating the summary, contributions and possible future research direction.

Chapter 2: Literature review

2.1 Overview

This chapter is dedicated to present an overview on state of the art in the field of cycle time prediction for milling free form surfaces and cycle time optimization by local toolpath smoothing for three-axis and five-axis machine tools. Cycle time prediction techniques available in the literature can be broadly classified into three categories (1) Cycle time prediction based on NC program, (2) Using material removal rates to forecast machining time and (3) Cycle time prediction considering machine characteristics and part program. Due to the inconsistent nature and unacceptable prediction error caused by first two approaches, most of the work in past decade is based on the third approach. Section 2.2 presents a more descriptive overview on various cycle time prediction techniques available in the literature. This is followed by Section 2.3 which covers the existing literature on toolpath smoothing techniques. Various global and local toolpath smoothing techniques are available in the literature. Although global smoothing techniques can be employed to reduce the machining time and generating smooth toolpath across given CAM points, the main disadvantage of this technique is the mathematical complexity involved to control and evaluate the exact fitting error. The challenge with the local smoothing approach however, is to ensure continuity at the junction between the corner rounding curve and the linear segment while respecting pre-defined error tolerance limit. Due to the relative maturity of three-axis local toolpath smoothing techniques, corner smoothing techniques that are present in literature for three-axis toolpath are reviewed first. This revision is extended to five-axis machine tools in the following subsection. The conclusions of the literature survey are presented in section 2.4.

2.2 Cycle time prediction for CNC milling operation

Cycle time prediction for free-form milling operation has received strong attention during past decades. Cycle time can be defined as the sum of primary cutting and auxiliary non-cutting motions for manufacturing a specific part. It depends upon a number of factors such as toolpath,

commanded feedrate, feedrate planning, geometry modification algorithm, CNC machine capabilities in terms of maximum achievable feedrate, acceleration, jerk, etc. Various cycle time prediction strategies were developed which can be broadly categorized into three groups i.e., (1) Using NC program information, (2) Using material removal rate and (3) Considering machine characteristics and part program.

2.2.1 Using NC program information

Conventionally, the machining time has been evaluated by NC code and improved by increasing commanded feedrate. The ordinary CAM software estimates the machining time considering that the feedrate is kept constant as set in the NC program ignoring the limitations of machine tool [1]. However, CNC machining centers are widely used for high-speed cutting operations nowadays with commanded feedrates as high as 20,000 mm/min [2]. A study of an NC program used for mold cutting shows that the error in the evaluation by a conventional facility exceeds 36% when the commanded feedrate is 3000 mm/min or higher and the average per-block feed distance is about 4mm [3]. This is because for high speed operations, the feedrate varies considerably along the toolpath due to mechanical (inertia) limitations of machine tool, feedrate profiling and limitations implemented by machine tool manufacturers to protect machine tool from damage during high-speed motion.

2.2.2 Using material removal rate

Othmani et al. [4] presented a method to calculate cutting time for complex geometry of contouring using linear or circular interpolation. The calculation method of cutting time is based on the ratio of the machined volume by the removed material rate during the contouring operation. This approach has been validated only for linear or circular interpolation. Hbaieb et al. [5] built an NC machining time forecasting model, using material removal rates to calculate rough machining time of a part in 2.5D milling operation. The proposed approach considered the roughing cutting time as the ratio of the pocket volume by removed material rate. The pocket is divided into volume distributed according to the real radial depth. However, the method is applicable only for pocket

milling operation. Jung [6] developed a feature based machining time estimation system for machined parts. For this, manufacturing features are classified in four categories i.e., (i) rotational features, (ii) prismatic features, (iii) slab features and, (iv) revolving features. Machining time corresponding to each feature is estimated based on material removal rates. It must be noted that none of the above mentioned geometry based methods were able to predict cycle time for all range of milling operations consistently as trajectory generation algorithm used in CNC system is completely ignored.

2.2.3 Using machine characteristics and part program

The cycle time prediction methods in this category can be further classified into three types. They are (1) Analytical evaluation, (2) Empirical evaluation and (3) Mechanistic evaluation.

(1) Analytical evaluation: In this approach, machine parameters are obtained from machine configuration files and are then used to simulate machine working conditions. Dugas et al. [7] developed a machining simulator which estimated real feedrate in two stages. In static look ahead stage, feedrate limits are calculated for a given toolpath and CNC controller. In dynamic look-ahead stage, feedrate profile is calculated based on given look ahead value assuming trapezoidal feedrate profile. Yan et al. [8] calculated cycle time based on factors such as command feedrate, block feed distance and feed acceleration/deceleration. Mou et al. [9] discovered the weakness of commercial software tools like VeriCut in terms of NC program machining simulation, analyzed the acceleration and deceleration principle of 840D Siemens numerical control system and developed a new NC program machining time simulation algorithm. So et al. [10] presented a machining time estimation algorithm based on feed angle, NC block processing time and ratio of translational to rotational motion for five-axis machine tools. However, the main disadvantage of analytical approach of time modeling is the need of the model parameters, such as drive time constants and gains. Such model parameters are set by the CNC manufacturer and might not be directly available to the machine tool user. The information about CNC implementation on the machines is also

difficult to get. It can be related to the acceleration profile, moving accuracy, and positional precision and it interferes directly with machines capacity to move at high speed. Even when suppliers specify the block processing time and the time required for the CNC to close its control loop, overall time required for machining a free-form path at high speed can depend on other parameters: the combination of these, and the way it was implemented on the machine controller. Even similar machines could have different processing capabilities, in terms of feed rate, since some features of the CNC can be set differently for different machine. All information required to develop an analytical model for machining time prediction can become quite difficult to obtain, which may lead to very limited results.

- (2) **Empirical evaluation**: Empirical models are based on linear regression models that fit a polynomial expression to a cloud of experimental data. However, the major disadvantage of this approach is the large number of the experiments needed to capture the phenomenon.
- (3) Mechanistic evaluation: Mechanistic models are the middle ground between analytical models and empirical models. The mechanistic approach utilizes the knowledge about the underlying mechanisms to propose models with some calibration constants. The values of these constants are found through experimentation. Monreal et al. [11] studied the influence of the toolpath strategy on the cycle time of machining products. A mechanistic approach based on the comparison between the ideal cycle time and the experimental cycle time was proposed. However, this approach was validated only for pocketing operation with zig-zag toolpath. Taking this work as background, Coelho et al. [11] proposed a mechanistic approach to identify real capacity of any CNC machine configuration. The method considered a variable called machining response time (MRT) which characterizes the real CNC machine is obtained by conducting a series of experiments with the machine. This method is suitable for machining time prediction of surfaces whose curvature radius changes greatly. However, the proposed a mechanistic approach for toolpath consisting linear segments. Siller et al. [12] proposed a mechanistic approach for more in high feedrates in greating whose curvature radius changes greatly. However, the proposed technique is valid only for toolpath consisting linear segments. Siller et al. [12] proposed a mechanistic approach for machine configure to a specific configure.

cycle time estimation in high speed milling of sculptured surfaces. The mechanistic model construction was based on: (a) the frequency distribution (histogram) of linear interpolation path lengths in the CNC program and (b) a characterization of the machine tool for brisk (large changes in tool path direction) and smooth movements (small changes in tool path direction). However, the proposed model resulted in upto 22% cycle time prediction error. Heo et al. [13] presented an NC machine time estimation model for machining sculptured surfaces, considering dynamic characteristics of the machine. The proposed estimation model used several factors, such as the distribution of NC blocks, angle between the blocks, federates, acceleration and deceleration constants, classifying tool feed rate patterns into four types based on the acceleration and deceleration profile, NC block length, and minimum feed rate. They are profiles having (i) only acceleration section, (ii) only deceleration section, (iii) acceleration and deceleration sections and (iv) acceleration, constant feedrate and deceleration sections. However, the lack of the measurement equipment or tools to gauge an exact minimum feed rate leads to an inaccuracy in the results. Moreover, it was assumed that acceleration limited feedrate profile is implemented on the CNC machine. This might not be the case considering various CNC controller manufacturers in the world.

From the above presented brief survey on available literature in the field on cycle time prediction, it can be concluded that none of the present approaches could predict cycle time for all range of milling operations within acceptable prediction error consistently because most of them ignore the effect of trajectory generation and corner smoothing algorithm implemented on the CNC system. To address this problem, a cycle time prediction technique based on part program, identified trajectory generation and corner smoothing algorithm and feedrate, acceleration and jerk limits of the drive is proposed in this thesis which can be applied for time modelling of all range of milling operations involving 2-axis, 3-axis and 5-axis machining within 7-8 % prediction error.

2.3 Geometrical toolpath smoothing

In the literature, several articles deal with the smoothness of motion along the toolpath. Considering machine tool axes have acceleration and jerk limitations, geometrical smoothing of the toolpath profile needs to be achieved first to ensure the smooth motion. Once geometry is smoothed, feedrate planning can be performed using jerk continuous trajectory planning to produce a smooth motion along the toolpath. Depending on the context, the geometrical toolpath smoothing techniques available in literature can be broadly categorised into two groups i.e., (1) Global smoothing and (2) Local smoothing. If a portion of the tool path is composed of a high density of short segments, it is possible to approximate all the segments by a curve. On the other hand, if the tool path is composed of long segments, it is important to follow precisely these segments and thus each transition has to be locally smoothed.

2.3.1 Global toolpath smoothing

Global smoothing has been widely studied in 3-axis and 5-axis cases. Different solutions can be used to smooth the orientation. Some of the suggested techniques are global toolpath smoothing using quaternions [14], drive movement smoothing [15], NURBS path interpolation [16] and decoupled position and orientation smooth curves [17] [18].

Ho et al. [14] proposed tool orientation smoothing technique using quaternion interpolation technique. For this, the initial representative tool orientations were assigned at crucial areas. This was followed by smooth interpolation of tool axis directions at general areas from the representative tool axis directions using the vector interpolation algorithm. In case interferences occurred, the representative tool axis directions were rearranged and new tool axis directions were interpolated. This process was repeated until all the tool axis directions passed the interference checking.

Beudaert et al. [15] proposed a toolpath smoothing algorithm based on drive constraints. Velocity, acceleration and jerk limits of each drive were employed to compute an evaluation of the maximum

reachable feedrate which was then used to localize the areas where the tool path had to be smoothed. These areas of high curvature were then smoothed iteratively in Machine Coordinate System (MCS) based on three parameters i.e., the tolerance on axis movement, the axis which has to be smoothed and the zone where it had to be smoothed. An N-buffer technique was also used to control the geometrical deviations on the part.

Lei et al. [14] presented non-uniform rational B-spline (NURBS) path interpolation method to generate smooth toolpath along discrete CAM points. Fleisig and Spence [17] interpolated the toolpath position and orientation with independent C^2 quintic B-Spline and quintic spherical Bezier spline respectively. To prevent feed fluctuations on the position and orientation, the quintic splines representing position and orientation are optimized using near arc length parameterization technique. Since the position and orientation spline may have different lengths, an orientation reparameterization spline is required to synchronize the interpolation of the position and orientation spline to a single parameter. The synchronizing spline is generated by assuming the path length as a linear chord, resulting in inaccuracies in orientations along highly curved path positions. Yuen et al. [18] presented a similar approach where the tool tip position and orientation was smoothed by fitting the position and orientation data to quintic splines in a decoupled fashion. The non-linear relationship between spline parameters and displacement along the path was approximated with ninth order and seventh order feed correction splines respectively.

It is evident that global smoothing techniques are widely employed to reduce the machining time and generating smooth toolpath across given CAM points. However, it is mathematically complex to control and evaluate the exact fitting error in global toolpath smoothing technique. Moreover, global smoothing techniques cannot be directly applied for local corner rounding because several requirements cannot be fulfilled. First the contour and orientation modifications have to be controlled locally and precisely avoiding oscillations. Second, the continuity at the junction between the newly introduced corner rounding curve and the linear segment needs to be ensured for local smoothing technique.

2.3.2 Local toolpath smoothing

Local toolpath smoothing also called as "corner smoothing" is widely used to generate smooth and non-stop motion along the toolpath. In this technique, polynomial curves or splines are inserted between two adjacent linear segments to blend them together. The challenge with the local smoothing approach is to ensure continuity at the junction between the corner rounding curve and the linear segment while respecting pre-defined error tolerance limit. Several techniques have been presented in the literature for three-axis corner smoothing. Some of them are presented as follows.

2.3.2.1 Three-axis corner smoothing

Local position smoothing has been widely studied for 3-axis machine tools. Indeed, different methods which give similar results are detailed in the literature. Some of the three-axis corner smoothing techniques available in literature are based on inserting cubic spline [19] [20] [21], quartic spline [22], quintic splines [23] [24] and conic smoothing segments [25] between adjacent linear toolpath segments.

A C^2 continuous non-uniform cubic B-Spline with eight control points is used by Pateloup et al. [19] to adapt to the linear and circular interpolation of the toolpath segments. Zhao et al. [20] developed a methodology for generation of curvature continuous cubic B-spline with five control points as transition curve. C^2 continuity along the toolpath and real-time performance are considered simultaneously in this approach. Bi et al. [21] used cubic Bezier curves defined by four control points to round the corner. The curvature radii of all transition curves were globally optimized to pursue the high feed speed by a linear programming model.

Yutkowitz and Chester [22] registered a patent with original assignee as Siemens Energy and Automation, Inc. describing a 3-axis corner rounding method based on inserting micro-spline between the adjacent linear toolpath segments. The inserted spline curve consists of two 4th order polynomial curves in parameter u. Near-arc length parameterization technique was utilised. The constraint equations are formulated to ensure the magnitude of the curvature vector of the modified 10

toolpath equals zero at the transition point between the linear toolpath and the inserted polynomial curve and it is maximum at the point of nearest approach to the sharp corner.

Sencer et al. [26] developed curvature-continuous smoothing scheme which fitted minimum curvature quintic B-splines to blend adjacent lines for three-axis tool motion. The proposed transition scheme ensured C^2 continuity transitions and a minimum curvature geometry delivering the fastest cycle time without violating the axis acceleration limits for 3-axis motion. The cornering error was controlled analytically allowing the user to set the desired cornering tolerance. Erkorkmaz et al. [23] also used a 5th order quintic spline curve to smoothen the sharp transitions in the toolpath. Two spline fitting strategies i.e., (1) Under-corner approach, and (2) Over-corner approach were developed for smoothening sharp corners. While the undercorner approach can be employed where the objective is to reduce the toolpath length, and therefore the cornering time, the over-corner approach which was based on stretching out the sharp corner with a smooth curve, can be mostly used where the 'undercut' caused by the large phase lag in low bandwidth servo controllers needs to be counteracted. Ernesto et al. [25] employed a Bezier conic and optimized the feedrate along the curve under acceleration bounds. Each corner was smoothed by replacing a subset of the path that contains it with a conic "splice" segment such that the newly inserted segment respected the user-defined error tolerance limit. The feedrate variation on this segment was determined by minimizing the traversal time, under the constraint of the axis acceleration bounds. The properties of Bernstein-form polynomials were then exploited to approximate the problem as a straight forward linear programming problem. Most of the existing three-axis corner smoothing techniques give a satisfying corner rounding curve in terms of respecting corner tolerance error and continuity or smoothness of toolpath. However, none of these papers tried to extend their solution to 5-axis tool path which brings new challenges.

2.3.2.2 Five-axis corner smoothing

While several local smoothing approaches are presented for three-axis toolpaths, there are still very limited works in the field of local toolpath smoothing for five-axis machine tools due to

difficulties related to smoothing of orientation. To the best of the author's knowledge, the only available work in the field of five-axis corner smoothing is presented by Beudaert et al. [27].

Buedaert et al. [27] presented five-axis corner smoothing method which was based on the representation of five-axis toolpath by a bottom curve defining the tooltip movement and by a top curve defining the tool orientation. Both the top and the bottom curves are smoothed separately to ensure second order acceleration continuity while respecting the user defined position and orientation tolerance limits. The spline parameter u_{bottom} of the bottom curve is found by arc length parameterization technique. An optimization problem is formulated to evaluate u_{top} to ensure smooth orientation connection. However, it was found that the proposed solution is very sensitive to even a tiny variation of the parameter which results in huge drop in feedrate. As the problem of five-axis corner smoothing is not well addressed in literature so far, a novel local toolpath smoothing technique is proposed in this thesis for five-axis machine tools where the objective is to achieve C^3 continuity at the junctions between the splines and the linear segments while respecting user-defined corner position tolerance (ε_{pos}) and orientation tolerance (ε_{ori}) limits.

2.4 Summary

An overview of existing literature in the field of cycle time prediction and toolpath smoothing for three-axis and five-axis machine tools are presented in this chapter. Cycle time prediction techniques available in the literature is classified into three categories (1) Cycle time prediction based on NC program, (2) Using material removal rates to forecast machining time and (3) Cycle time prediction considering machine characteristics and part program. Due to the inconsistent nature and unacceptable prediction error caused by first two approaches, most of the work in past decade is based on the third approach which can be further classified into three categories i.e., (1) Analytical evaluation, (2) Empirical evaluation and (3) Mechanistic evaluation of cycle time. The main disadvantage of analytical approach of time modeling is the need of the model parameters, such as drive time constants and gains. Such model parameters are set by the CNC controller manufacturer and might not be directly available to the machine tool user.

which are based on linear regression models that fit a polynomial expression to a cloud of experimental data needs large number of the experiments to capture the phenomenon. However, the mechanistic models which are middle ground between analytical and empirical models are most commonly used for time modeling of milling operations. However, the mechanistic approaches available in literature could not predict cycle time for all range of milling operations within acceptable prediction error consistently. To address this problem, a cycle time prediction technique is proposed in this thesis which can be applied for time modelling of all range of milling operations involving 2-axis, 3-axis and 5-axis machining within 7-8 % prediction error.

Toolpath smoothing techniques available in the literature can be broadly classified into two categories (1) Global toolpath smoothing, (2) Local toolpath smoothing. Although global smoothing techniques can be employed to reduce the machining time and generating smooth toolpath across given CAM points, the main disadvantage of this technique is the mathematical complexity involved to control and evaluate the exact fitting error. The challenge with the local smoothing approach however, is to ensure continuity at the junction between the corner rounding curve and the linear segment while respecting pre-defined error tolerance limit. While several local smoothing approaches are presented for three-axis toolpaths, there are still very limited works in the field of local toolpath smoothing for five-axis machine tools due to difficulties related to smoothing of orientation. To address this problem, a novel local toolpath smoothing technique is proposed in this thesis for five-axis machine tools where the objective is to achieve C^3 continuity at the junctions between the splines and the linear segments while respecting user-defined corner position tolerance (ε_{pos}) and orientation tolerance (ε_{ori}) limits.

Chapter 3: Cycle time prediction for milling process

3.1 Overview

The production scheduling for machining process involves the sequence of operations and their input times along with other major decision factors, in order to avoid overloading a machine with multiple assignments at the same time. It is not an easy task to design an exact production schedule due to unexpected demands and complexity of the process, namely modification and reproduction. The actual production schedules are customarily modified by a production planning engineer who co-ordinates the whole working process and schedule it in real time [13]. However, knowing the exact machining time in advance can lead to more accurate production process planning which in turn, greatly affects the performance of a shop floor from production economics perspective. Besides, machining time is one of the most important metric to analyze CNC machine tool capabilities. It can also be used by product designers to improve and optimize the design of products and to find optimal tool path strategy to machine a component.

Conventionally, the machining time has been evaluated by NC code and improved by increasing commanded feedrate. This has worked well for conventional low-speed machining process. However, for high speed operations, the feedrate varies considerably along the toolpath due to mechanical (inertia) limitations of machine tool and limitations implemented by machine tool manufacturers to protect machine tool from damage during high-speed motion which leads to higher machining time prediction error.

In this chapter, a cycle time prediction algorithm based on various factors including NC program, feedrate planning, geometry modification algorithm and machine characteristics is proposed for accurate estimation of machining time for milling free form surfaces. For this, simple sets of experiments have been conducted in Mori Seiki NMV5000 DCG commercial CNC machine to identify the implemented trajectory generation and corner smoothing techniques used, and feed

drive dynamics of the CNC system as described in section 3.2. An analytical approach for predicting cycle time based on the identified CNC system dynamics and given part program is presented in section 3.3. In the last section, the experimental and predicted feedrate profiles and cycle time are compared for various 2-axis, 3-axis and 5-axis toolpaths to validate the proposed cycle time estimation strategy.



Figure 3.1 Flow of commands in Commercial CNC System

3.2 Modeling dynamics of CNC systems

Flow of commands in a CNC machining process is shown in figure 3.1. CAD/ CAM system generates reference toolpath in the form of part program which is loaded into the CNC machines where each block passes through geometry modification and trajectory generation algorithm. Here,

discrete displacement commands are generated for each axes at every control interval. These drive commands are passed on to closed-loop feed-drive system for each axis where the control law shapes the final drive positions.

Most of the commercial CNC machine can be operated in exact stop mode (G61) and normal cutting mode (G64) [28]. In exact stop mode, the toolpath motion ceases at each block transition points as shown in figure 3.2. In this case, geometry modification module is deactivated and inposition status is checked for each block. On the other hand, the toolpath planning for cutting/ continuous mode is done to ensure smooth path transitions between consecutive blocks containing path movements which results in rounded corners as shown in figure 3.2.



Figure 3.2 Feedrate profile for exact-stop and continuous mode

To forecast NC cycle time accurately in exact stop mode, "real" feedrate needs to be predicted which depends on multiple factors including programmed feedrate, length of toolpath blocks, directional changes in the toolpath, machine performance parameters such as feedrate, acceleration and jerk limits of the drives, feedrate interpolation method and closed loop feed-drive dynamics. In cutting/ continuous mode, the NC cycle time depends on corner smoothing technique in addition to all the above-mentioned factors which affect cycle time in exact stop mode.



Figure 3.3 Configuration of Mori Seiki NMV5000 DCG commercial machining center [29]

NC part program contains programmed feedrate and toolpath related information such as block length, directional changes in toolpath, etc. The feedrate, acceleration and jerk limits of the drives are often provided by the manufacturer. Most modern CNC machines also provide users with feedrate profile options such as linear and exponential feedrate profile in the acceleration/ deceleration range [13]. However, the parameters related to the feedrate profiles, corner smoothing technique and closed-loop feed drive dynamics are not revealed to end users which need to be identified for accurate prediction of machining time.

For identification purpose, simple sets of experiments are performed on NMV5000 DCG vertical machining center by Mori Seiki Co. Ltd. This CNC machine has Fanuc 30i controller and five-axis kinematic configuration with B/C axis rotary table [30] as shown in figure 3.3. Both B and C axes are driven by direct drive (DD) motors and has rotation range of 340 and 360 degrees respectively. Each of the Y and Z axes are driven by two synchronously controlled ball-screw drives and have axis travel range of 510 mm. However, X-axis is driven by a single ball-screw drive with axis travel range of 730 mm. Part program (G-code) is fed to the machine using Mapps IV interface [31] and experimental data is measured online at a sampling time, $t_s = 1 ms$ using Fanuc servo guide software (version 6.00) installed on a laptop which is connected to the NC system using ethernet cable.

3.2.1 Identification of trajectory parameters

For identification of trajectory parameters, the machine tool is commanded to move in linear toolpath segments (G01 command) of varying length at a range of programmed feedrates. The high-precision contour control mode is turned off as it activates various sub-programs which results in modification of feedrate profile generated by the trajectory generation module of CNC machine tool. The experiments are carried out in exact-stop mode to avoid the effect of smoothing technique in trajectory generation process. The experimental conditions are summarized below:

- Block length: 1, 2, 5, 10, 50 and 100 mm
- Commanded feedrate: 1000, 2000, 3000, 4000, 5000 and 6000 mm/min
- High-precision contour control: off



• Mode: Exact stop (G61)

Figure 3.4 Kinematic profiles of Mori Seiki NMV5000

Feedrate data $f(t_i)$ is measured using servo guide software at discrete time steps $t_i \forall i = 1, 2, ..., n$ such that $(t_{i+1} - t_i) = t_s$ and $t_n = t_1 + t_2 + t_3$ (see figure 3.4). Acceleration $a(t_i)$ and jerk $j(t_i)$ along the toolpath can be calculated on a fly using digital differentiation technique.

$$a(t_{i}) = \frac{f(t_{i}) - f(t_{i-1})}{t_{s}} \forall i = 2, 3, ..., n;$$

$$j(t_{i}) = \frac{a(t_{i}) - a(t_{i-1})}{t_{s}} = \frac{f(t_{i}) + f(t_{i-2}) - 2f(t_{i-1})}{(t_{s})^{2}}, \quad \forall i = 3, 4, ..., n;$$
(3.1)

Figure 3.4 shows the pattern of feedrate, acceleration and jerk profiles of Mori Seiki NMV5000 DCG as measured by servo guide software. For further investigation, the acceleration, constant feedrate and deceleration sections of the feedrate profile are analyzed separately. As NMV5000 DCG machine is set to follow the exponential feedrate profile [32], the initial model of the feedrate profile for acceleration section is chosen as

$$f(t) = Ae^{-t/T_1} + Be^{-t/T_2} + C$$
(3.2)

To estimate the values of the unknown parameters A, B, C, T_1 and T_2 , an unconstrained optimization problem is formulated where the objective is to find the optimal value of (5×1) vector $\mathbf{x} = [A, B, C, T_1, T_2]^T$ such that the scalar cost function $c(\mathbf{x})$ which can be expressed as sum of square of residuals is minimized.

$$\min_{\mathbf{x}} c(\mathbf{x}) = \sum_{i=1}^{m} \left\{ f(t_i) - \left(A e^{-\frac{t_i}{T_1}} + B e^{-\frac{t_i}{T_2}} + C \right) \right\}^2$$
(3.3)

Where $t_m = t_1$ is the acceleration time (see Fig 3.4). For this, trust region non-linear least-square optimization technique as explained in [33], [34], [35] is used. It is a global convergence algorithm. The first step is to specify the initial guess $\mathbf{x}_0 = [A_0, B_0, C_0, T_{10}, T_{20}]^T$ and initial trust-region size Δ_0 . Trust-region size can be defined as the radius of sphere with \mathbf{x}_k as center where the solution can lie. At any iteration k, the value of the step size $\mathbf{p}_k = [\Delta A_k, \Delta B_k, \Delta C_k, \Delta T_{1k}, \Delta T_{2k}]^T$ is calculated. It should be ensured that modulus of step size \mathbf{p}_k must be less than the trust-region size Δ_k at the k^{th} iteration. The optimal value of \mathbf{p}_k which minimizes cost function is given by: $\mathbf{p}_k = -(H(\mathbf{x}_k) + \lambda I)^{-1}g(\mathbf{x}_k)$ (3.4)

20

where $g(x_k)$ is (5×1) vector which represents the gradient of cost function c(x) at the current iteration point x_k . It can be formulated as

$$g(\boldsymbol{x}_{k}) = \frac{\partial c}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{x}=\boldsymbol{x}_{k}} = \left(\left[\frac{\partial c}{\partial A} \frac{\partial c}{\partial B} \frac{\partial c}{\partial C} \frac{\partial c}{\partial T_{1}} \frac{\partial c}{\partial T_{2}}\right]^{T}\right)\Big|_{\boldsymbol{x}=\boldsymbol{x}_{k}},$$
(3.5)

 $H(x_k)$ is (5 × 5) symmetric matrix of second derivatives (Hessian Matrix) of the cost function c(x) at x_k . It can be expressed as

$$H(\mathbf{x}_{k}) = \begin{bmatrix} \frac{\partial^{2} c}{\partial A^{2}} & \cdots & \frac{\partial^{2} c}{\partial A \partial T_{2}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^{2} c}{\partial T_{2} \partial A} & \cdots & \frac{\partial^{2} c}{\partial T_{2}^{2}} \end{bmatrix} \Big|_{\mathbf{x}=\mathbf{x}_{k}}$$
(3.6)

and λ is a positive scalar value such that

$$\|\boldsymbol{p}_{k}\| = \left\| -(H(\boldsymbol{x}_{k}) + \lambda I)^{-1} g(\boldsymbol{x}_{k}) \right\| \le \Delta_{k}$$
(3.7)

If the trust region size Δ_k shrinks to less than 10^{-3} , it implies the current solution can be trusted and the algorithm terminates. Otherwise, the value of x_k and trust-region size Δ_k needs to be updated for next iteration. For this, reduction ratio red_k is evaluated.

$$red_k = \frac{c(\boldsymbol{x}_k) - c(\boldsymbol{x}_k + \boldsymbol{p}_k)}{m_k(\boldsymbol{0}) - m_k(\boldsymbol{p}_k)}$$
(3.8)

where,

$$m_k(\boldsymbol{p}) = c(\boldsymbol{x}_k) + \boldsymbol{p}^T g(\boldsymbol{x}_k) + \frac{1}{2} \boldsymbol{p}^T H(\boldsymbol{x}_k) \boldsymbol{p}$$
(3.9)

If the reduction ratio is large, the trust-region size is increased in the next iteration. If the reduction ratio is too small, the trust-region size is decreased in the next iteration. Otherwise, it is kept the same.

$$\Delta_{k+1} = \begin{cases} \frac{1}{4} \Delta_k & \text{if } red_k < \frac{1}{4} \\ \min(2\Delta_k, \Delta_{max}) & \text{if } red_k > \frac{3}{4} \\ \Delta_k & \text{otherwise} \end{cases}$$
(3.10)

Furthermore, a step p_k will only be accepted if the reduction ratio is not too small.

$$\boldsymbol{x_{k+1}} = \begin{cases} \boldsymbol{x_k} + \boldsymbol{p_k} & \text{if } red_k > 1/8\\ \boldsymbol{x_k} & \text{otherwise} \end{cases}$$
(3.11)

As Trust-region is a global optimization technique, the initial guess \mathbf{x}_0 does not have much effect on the final result. For our purpose, the initial guess \mathbf{x}_0 is chosen as $[10,10,10,0.1,0.2]^T$. The maximum step length Δ_{max} and initial trust-region size Δ_0 are chosen as 100 and 10, respectively. For identification of trajectory profile, tool is commanded to move 100 mm in x-direction at commanded feedrate of 6000 mm/min and feedrate data is collected at a sampling time of 1 *ms*. The data corresponding to the acceleration region only is used for identification purpose. The trustregion algorithm converges in 21 iterations in this case. The plot showing the cost function and trust-region size in each iteration are shown in figure 3.5 and 3.6 respectively. The optimal value of unknown parameters in this case are found as $\hat{A} = 207.25 \text{ mm/sec}$, $\hat{B} = -304.14 \text{ mm/}$ sec, $\hat{C} = 99.96 \text{ mm/sec}$, $\hat{T}_1 = 0.033 \text{ s}$ and $\hat{T}_2 = 0.049 \text{ s}$.

Next, the procedure is repeated for other five sets of experiments in which tool is commanded to move 100 mm in x-direction at commanded feedrate of 1000, 2000, 3000, 4000 and 5000 mm/min respectively. Trust-region algorithm is applied to the collected feedrate data for all the five cases separately to estimate the values of the parameters A, B, C, T_1 and T_2 . The result of trust region optimization is tabulated in table 3.1.



Figure 3.5 Convergence of cost function in Trust-region algorithm



Figure 3.6 Convergence of Trust-region size in Trust-region algorithm

Commandad faadrata	Values of estimated parameters				
(f) (mm/sec)	Â	\widehat{B}	Ĉ	\widehat{T}_1	\widehat{T}_2
(J_c) (mm/sec)	(mm/ sec)	(mm/sec)	(mm/sec)	(sec)	(sec)
16.67	36.8	-53.65	16.5	0.033	0.049
33.33	67.77	-101.06	33.2	0.0325	0.048
50	104.12	-155.13	48.6	0.033	0.05
66.67	137.5	-207.17	67	0.0332	0.0481
83.33	168.2	-254.8	82.8	0.033	0.049
100	207.25	-304.14	99.96	0.033	0.049

Table 3.1 Trust region parameter estimates

From trust-region parameter estimates, it is found that the values of time constants \hat{T}_1 and \hat{T}_2 , estimated as 0.033 and 0.049 seconds respectively are independent of the commanded feedrate and constant for a machine. However, the values of other parameters vary. To express the parameters \hat{A}, \hat{B} and \hat{C} as functions of commanded feedrate (f_c) , general linear model of regression analysis is used. For regression analysis, an over determined system is considered [36] as:-

$$\sum_{j=1}^{n} X_{ij} \beta_j = y_i \ \forall \ i = 1, 2, \dots, m$$
(3.14)

where *m* linear equations are used to find *n* **unknown** coefficients, $\beta_1, \beta_2, ..., \beta_n$, with m > n. It should be noted that X_{ij} and y_i are **known** inputs and outputs respectively. Equation (3.14) can be written in matrix form as $X\beta = y$,

where $\boldsymbol{X} = \begin{bmatrix} X_{11} & \cdots & X_{1n} \\ \vdots & \ddots & \vdots \\ X_{m1} & \cdots & X_{mn} \end{bmatrix}, \boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_m \end{bmatrix}, \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$

24

(3.15)

Such a system of equations are solved by forming a minimization problem where the objective is to minimize the sum of square of errors, $S(\beta)$ which is formulated as

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{m} (e_i)^2 = \sum_{i=1}^{m} \left(y_i - \sum_{j=1}^{n} X_{ij} \beta_j \right)^2$$
(3.16)

 $S(\beta)$ is minimized by equating its gradient value to zero which gives the estimated value of the unknown parameters $\hat{\beta}$

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y}$$
(3.17)

To express \widehat{A} , \widehat{B} and \widehat{C} as functions of commanded feedrate (f_c) , it is assumed that

$$\widehat{\boldsymbol{A}} = \mu_1 \boldsymbol{f}_c, \qquad \widehat{\boldsymbol{B}} = \mu_2 \boldsymbol{f}_c, \qquad \widehat{\boldsymbol{C}} = \mu_3 \boldsymbol{f}_c \tag{3.18}$$

where \hat{A} , \hat{B} , \hat{C} and f_c are (5 × 1) vectors as shown in table 1. Using linear regression, the unknown parameters μ_1 , μ_2 and μ_3 are computed as

$$\widehat{\mu_{1}} = (f_{c}^{T} f_{c})^{-1} f_{c}^{T} \widehat{A} = 2.057243 \approx \frac{-T_{1}}{T_{1} - T_{2}}$$

$$\widehat{\mu_{2}} = (f_{c}^{T} f_{c})^{-1} f_{c}^{T} \widehat{B} = -3.06615 \approx \frac{T_{2}}{T_{1} - T_{2}}$$

$$\widehat{\mu_{3}} = (f_{c}^{T} f_{c})^{-1} f_{c}^{T} \widehat{C} = 0.99574 \approx 1$$
(3.19)

This brings us to the following results:

$$\hat{A} \approx \frac{-f_c T_1}{T_1 - T_2}, \quad \hat{B} \approx \frac{f_c T_2}{T_1 - T_2}, \quad \hat{C} \approx f_c$$
(3.20)

From equations (3.2) and (3.20), the feedrate equation corresponding to the acceleration section of the profile can be expressed as

$$f(t) = \left(\frac{-f_c T_1}{T_1 - T_2}\right) e^{-t/T_1} + \left(\frac{f_c T_2}{T_1 - T_2}\right) e^{-t/T_2} + f_c \quad \forall \ 0 \le t < t_1$$
(3.21)

25
where t_1 is the acceleration time as shown in figure 3.4. Similarly, the feedrate equation corresponding to the deceleration section is formulated. The identified feedrate equation along the toolpath can be expressed as

$$f(t) = \begin{cases} \frac{-f_c}{T_1 - T_2} \left(T_1 e^{-\frac{t}{T_1}} - T_2 e^{-\frac{t}{T_2}} \right) + f_c, & 0 \le t < t_1 \\ f_c, & t_1 \le t < t_1 + t_2 \\ \frac{f_c}{T_1 - T_2} \left(T_1 e^{-\frac{(t - t_1 - t_2)}{T_1}} - T_2 e^{-\frac{(t - t_1 - t_2)}{T_2}} \right), & t_1 + t_2 \le t \le t_1 + t_2 + t_3 \end{cases}$$
(3.22)

Differentiating equation (3.22) with respect to time t, the acceleration profile a(t) is expressed as,

$$a(t) = \begin{cases} \frac{f_c}{T_1 - T_2} \left(-e^{-\frac{t}{T_1}} + e^{-\frac{t}{T_2}} \right), & 0 \le t < t_1 \\ 0, & t_1 \le t < t_1 + t_2 \\ \frac{f_c}{T_1 - T_2} \left(-e^{-(t - t_1 - t_2)/T_1} + e^{-(t - t_1 - t_2)/T_2} \right), & t_1 + t_2 \le t \le t_1 + t_2 + t_3 \end{cases}$$
(3.23)

Again, differentiating equation (3.23) with respect to time, the jerk profile j(t) is obtained as,

$$j(t) = \begin{cases} \frac{f_c}{T_1 - T_2} \left(\frac{1}{T_1} e^{-\frac{t}{T_1}} - \frac{1}{T_2} e^{-\frac{t}{T_2}} \right), & 0 \le t < t_1 \\ 0, & t_1 \le t < t_1 + t_2 \\ \frac{f_c}{T_1 - T_2} \left(\frac{1}{T_1} e^{-(t - t_1 - t_2)/T_1} - \frac{1}{T_2} e^{-(t - t_1 - t_2)/T_2} \right), & t_1 + t_2 \le t \le t_1 + t_2 + t_3 \end{cases}$$
(3.24)

It should be noted that experiments have been conducted for different toolpath lengths under three circumstances: (1) linear: movement along one specific axis (X; Y; Z); (2) planar: toolpath involving two axes (XY; YZ; XZ); (3) spatial: three axes and five axes toolpaths. Experimental data reveals that the trajectory generation algorithm does not change as toolpath length or number of axis in movement changes.

3.2.2 Prediction of corner smoothing technique

For the prediction of corner smoothing technique implemented on the CNC system, a simple Gcode command is sent to the system and the machine is commanded to operate in exact-stop mode. Feedrate data is measured using servo guide software.

<u>G-Code command:</u> G61; G01 X4.455 F1000.; X2.327 F500.; X7.9 F2000.; X-4.55 F1000.;

Then, the experiment is repeated in the continuous mode and feedrate data is recorded again. Figure (3.7) shows the experimental feedrate data for exact-stop and continuous mode.



Figure 3.7 Experimental feedrate profile in (i) Exact-stop mode, (ii) Continuous mode



Figure 3.8 Predicted feedrate profile in Continuous mode

After carefully analyzing the feedrate data, it was predicted that unlike exact-stop mode, the execution of the next block starts as the feedrate of the current block starts decelerating in continuous mode as shown in the figure (3.8). Then, feedrate contributions by all the blocks at a specific time are added to compute the feedrate profile along the toolpath in continuous mode. The predicted feedrate profile in continuous mode is then compared with the experimental feedrate profile as discussed in section (3.4) to validate the prediction of corner smoothing technique.

3.2.3 Identification of closed-loop feed-drive dynamics

Commercial CNC drive control systems have generally closed and propriety architecture and a detailed dynamic model including the controller is not provided to the users [37]. Therefore, an identification technique is required to estimate drive dynamics so that machine behavior can be simulated in the virtual environment accurately. Erkorkmaz et al [38] and Sencer et al [37] proposed a rapid closed loop identification technique by conducting a short G-code test. However, it is difficult to excite commercial CNC machines by G-code command leading to inaccuracy in measured FRF.

In this section, a technique for identification of closed loop feed drive dynamics of NMV5000 DCG CNC system is presented where the closed loop feed drive system is excited by sending disturbance torque commands. The control block diagram of NMV5000 DGC feed-drive system is shown in figure 3.9. For identification of unknown velocity loop transfer function, the feed-forward block is deactivated. Position gain is set as 1 to prevent position loop effect.



Figure 3.9 Control block diagram of Mori Seiki NMV5000 feed-drive system

The velocity loop is then excited by sending disturbance torque in the frequency range of 1 to 1000 [*rad/sec*] and the output torque is measured using servo guide software. Having known the input and the output torque data, experimental frequency response function (FRF) of the velocity loop is computed by dividing the output response by the input command. The experimental velocity loop FRF of X-drive as measured by servo guide is shown in figure 3.10. From the experimental data, it is observed that two notch filters are implemented in the system at 200 Hz and 360 Hz respectively to block the harmonic component of the input command that coincides with the resonance frequency of the machine. The resonance elimination (RE) filters are designed based on the parameters provided in the manual [39]. The designed notch filters are

$$RE1(s) = \frac{s^2 + 104.08s + 4\pi^2(200)^2}{s^2 + 346.93s + 4\pi^2(200)^2},$$

$$RE2(s) = \frac{s^2 + 4\pi^2(360)^2}{s^2 + 188.4956s + 4\pi^2(360)^2}$$
(3.25)
29



Figure 3.10 Experimental velocity loop FRF of X-drive of Mori Seiki NMV5000 DCG CNC system

Notch filters are ignored for drive identification process. The drive parameters are then estimated using continuous time model identification approach presented by Pintelon [40] [41] from the complex frequency response data $\tilde{G}(\omega_k)$.

$$\breve{G}(\omega_k) = M_k e^{j\varphi_k} = R_k + jI_k \tag{3.26}$$

where M_k and φ_k represent experimentally measured magnitude and phase corresponding to angular frequency ω_k for k = 1, 2, ..., N. The frequency range 5 to 150 Hz (~10 to 950 [*rad/sec*]) is used for identification purpose as notch filters do not affect the data in this frequency range. The initial closed loop feed drive system is assumed as

$$G(s,\boldsymbol{\theta}) = \frac{B(s,\boldsymbol{\theta})}{A(s,\boldsymbol{\theta})} = \frac{b_0 + b_1 s + \dots + b_m s^m}{a_0 + a_1 s + \dots + a_n s^n} \quad \forall \ a_n = 1$$
(3.27)

where *s* is Laplace operator and $\boldsymbol{\theta} = [a_{n-1} \dots a_0 \ b_m \dots \ b_0]^T$. The objective is to estimate the parameter vector $\boldsymbol{\theta}$ of the continuous time plant model described in equation (3.27) from *N* sampled measurements of the angular frequencies and system response data $\mathbf{Z}^N = \{\omega_k; \boldsymbol{G}(\omega_k)\}_{k=1}^N$. It is assumed that the orders *n* and *m* are known in advance. For simplicity, let us assume that the values of *n* and *m* are 2 and 1 respectively. Then, the closed loop plant model can be expressed as

$$G(s, \theta) = \frac{B(s, \theta)}{A(s, \theta)} = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0}; \theta = [a_1 \ a_0 \ b_1 \ b_0]^T$$
(3.28)

The identification problem implies that we want to find $\hat{\theta}$ such that it minimizes the cost function

$$c(\boldsymbol{\theta}) = \sum_{k=1}^{N} \left| \breve{G}(\omega_k) - G(j\omega_k, \boldsymbol{\theta}) \right|^2$$
(3.29)

Thus,

$$\widehat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} \sum_{k=1}^{N} \left(\frac{\left| A(j\omega_k, \boldsymbol{\theta}) \breve{G}(\omega_k) - B(j\omega_k, \boldsymbol{\theta}) \right|^2}{|A(j\omega_k, \boldsymbol{\theta})|^2} \right)$$
(3.30)

It should be noted that the above minimization problem is non-linear in nature as the parameter $\boldsymbol{\theta}$ which needs to be estimated comes in denominator. To linearize the problem, a known value $A^*(\omega_k)$ is substituted for $A(j\omega_k, \boldsymbol{\theta})$ term that occurs in denominator.

$$\widehat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} \sum_{k=1}^{N} \left(\frac{\left| A(j\omega_k, \boldsymbol{\theta}) \breve{G}(\omega_k) - B(j\omega_k, \boldsymbol{\theta}) \right|^2}{|A^*(\omega_k)|^2} \right)$$
(3.31)

Substituting the values of $A(j\omega_k, \theta)$, $B(j\omega_k, \theta)$ and $\breve{G}(\omega_k)$, we get

$$\widehat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} \sum_{k=1}^{N} \left(\frac{\left| (R_k + jI_k)(\omega_k)^2 - \left\{ \left((-I_k + jR_k)\omega_k \right) a_1 + (R_k + jI_k)a_0 - j\omega_k b_1 - b_0 \right\} \right|^2}{|A^*(\omega_k)|^2} \right)$$
(3.32)

$$\Rightarrow \widehat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} \sum_{k=1}^{N} |Z_k - \boldsymbol{X}_k \boldsymbol{\theta}|^2$$

where

$$Z_{k} = \frac{(R_{k} + jI_{k})(\omega_{k})^{2}}{A^{*}(\omega_{k})}; X_{k} = \left[\frac{(-I_{k} + jR_{k})\omega_{k}}{A^{*}(\omega_{k})} \quad \frac{R_{k} + jI_{k}}{A^{*}(\omega_{k})} \quad \frac{-j\omega_{k}}{A^{*}(\omega_{k})} \quad \frac{-1}{A^{*}(\omega_{k})}\right]$$
(3.33)

The solution of the above least square problem can be evaluated as

$$\widehat{\boldsymbol{\theta}} = (\overline{\boldsymbol{X}}'\boldsymbol{X})^{-1}\overline{\boldsymbol{X}}'\boldsymbol{Z}; \ \boldsymbol{Z} = \begin{bmatrix} \boldsymbol{Z}_1 \\ \boldsymbol{Z}_2 \\ \vdots \\ \boldsymbol{Z}_N \end{bmatrix}; \ \boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \\ \vdots \\ \boldsymbol{X}_N \end{bmatrix}$$
(3.34)

where \overline{X}' represents conjugate transpose of the X matrix. It should be noted here that some of the elements of Z vector and X matrix are complex numbers. Thus, it is highly probable that the estimated value of coefficient vector $\widehat{\theta} = [\widehat{a_1} \ \widehat{a_0} \ \widehat{b_1} \ \widehat{b_0}]^T$ will be complex numbers as well. This can be avoided by adding negative frequencies. It can be shown that if $\check{G}(\omega_k) = R_k + jI_k$, then the value of $\check{G}(-\omega_k) = R_k - jI_k$. If for all the positive frequencies, the negative frequencies are also included, it is observed that the optimization problem gives real valued parameters. The initial estimate of denominator term $A^*(\omega_k)$ is assumed as 1 for all k. $\widehat{\theta} = [\widehat{a_1} \ \widehat{a_0} \ \widehat{b_1} \ \widehat{b_0}]^T$ is estimated using the least square optimization method as formulated in equation (3.33) and (3.34). Based on the estimated values of $\widehat{a_1}$ and $\widehat{a_0}$, the value of denominator term is updated as

$$A^*(\omega_k) = A(j\omega_k, \widehat{\theta}) = (\widehat{a_0} - \omega_k^2) + j(\widehat{a_1}\omega_k)$$
(3.35)

Using updated denominator value, $\hat{\theta}$ is estimated again using least square optimization technique. This method is repeated until following convergence criteria is fulfilled for a small value ε .

$$\left\|\widehat{\boldsymbol{\theta}}_{i} - \widehat{\boldsymbol{\theta}}_{i-1}\right\| \le \varepsilon \tag{3.36}$$



Figure 3.11 Experimental and identified closed loop frequency response function of X-axis

For the closed-loop identification of X-drive, the order of numerator and denominator of the transfer function is assumed as 3 and 4 respectively. Based on the described least square

optimization method, the identified closed loop FRF (without notch filter) of X-drive is estimated as

$$G_{cl}(s) = \frac{-3.288s^3 + 3.247e5s^2 + 1.036e8s + 4.08e10}{s^4 + 523.3s^3 + 8.381e5s^2 + 1.77e8s + 4.08e10}$$
(3.37)

After that, the notch filters as designed in (3.25) were multiplied with $G_{cl}(s)$ to find the closed loop response of the x-drive. The identified and experimental FRF of the x-drive are shown in figure 3.11. Similarly, closed-loop identification of Y, Z, B and C drives are carried out and results are presented in table 3.2 and figure 3.12.

 Table 3.2 Identified closed loop feed-drive dynamics of Mori Seiki NMV5000 CNC system

 Resonance Elimination Filter, RE(s) Closed loop transfer function, $G_{cl}(s)$

	Resonance Elimination Filter, <i>RE</i> (<i>s</i>)	Closed loop transfer function, $G_{cl}(s)$
X-drive	$\frac{s^{2} + 104.08s + 4\pi^{2}(200)^{2}}{s^{2} + 346.93s + 4\pi^{2}(200)^{2}},$ $\frac{s^{2} + 4\pi^{2}(360)^{2}}{s^{2} + 188.4956s + 4\pi^{2}(360)^{2}}$	$\frac{-3.288s^3 + 3.247e5s^2 + 1.036e8s + 4.08e10}{s^4 + 523.3s^3 + 8.381e5s^2 + 1.77e8s + 4.08e10}$
Y-drive	$\frac{s^2 + 6.2838s + 4\pi^2(390)^2}{s^2 + 628.38s + 4\pi^2(390)^2},$ $\frac{s^2 + 4\pi^2(260)^2}{s^2 + 188.4956s + 4\pi^2(260)^2}$	$\frac{-190.8s^3 + 6.126e5s^2 + 2.694e7s + 4.827e10}{s^4 + 894.9s^3 + 9.331e5s^2 + 1.143e8s + 5.59e10}$
Z-drive	$\frac{s^2 + 6.2838s + 4\pi^2(530)^2}{s^2 + 628.38s + 4\pi^2(530)^2}$	$\frac{-375s^3 + 1.668e6s^2 + 7.16e8s + 9.608e9}{s^4 + 2365s^3 + 2.645e6s^2 + 1.938e9s + 9.03e9}$
B-drive	-	$\frac{346.9s^4 - 1.458e6s^3 + 2.227e9s^2 + 4.219e11s + 5.02e12}{s^5 + 9155s^4 + 6.47e6s^3 + 3.87e9s^2 + 7.535e11s + 4.597e12e10}$
C-drive	-	$\frac{26.44s^3 + 5.44e4s^2 + 1.66e7s + 5.273e9}{s^4 + 306s^3 + 2.04e5s^2 + 2.384e7s + 5.319e9}$



Figure 3.12 Experimental and identified closed loop FRF

3.3 Cycle time estimation

After including the effect of the identified trajectory generation algorithm, corner smoothing technique and feed-drive dynamics in the prediction of cycle time, it is observed that the cycle time of NC machining process is predominantly depended upon trajectory generation and corner smoothing techniques implemented on the CNC system. The closed-loop feed drive dynamics however, does not have much influence on the cycle time. Thus, the effect of closed loop feed drive dynamics is ignored for the cycle time estimation. The schematic diagram of the proposed algorithm for cycle time estimation is shown in figure 3.11, with a brief description as follows.

To predict the cycle time for a given toolpath, the acceleration, constant feedrate and deceleration time corresponding to each block of NC part program are calculated. Then, cycle time for the given toolpath is evaluated based on mode (Continuous or exact stop) chosen by the user. The default mode of operation for NMV5000 DCG CNC machine is units per minute feedrate mode in which an "F" word on NC block is interpreted to mean the controlled point should move at a certain number of millimeters per minute or degrees per minute, depending upon which axis or axes are moving [39]. In this case, to predict the feedrate along the toolpath, pseudo-distance l_i is calculated based on increments on the axis commands.

$$l_{i} = \left(\Delta X_{i}^{2} + \Delta Y_{i}^{2} + \Delta Z_{i}^{2} + \Delta B_{i}^{2} + \Delta C_{i}^{2}\right)^{1/2} \quad \forall i = 1, 2 \dots, N$$
(3.44)

Where *i* represents current block number of NC part program and *N* is the total number of blocks in the given NC program. ΔX_i , ΔY_i and ΔZ_i are the incremental translational axis commands in [mm] and ΔB_i and ΔC_i are incremental rotary axis commands in [degrees] at *i*th block. Initially, it is assumed that the pseudo block length l_i for the *i*th NC block is long enough to allow the tool to reach the commanded feedrate f_{ci} . As shown in figure 3.4, the tool reaches commanded feedrate f_{ci} at the end of acceleration section. Thus,

$$f_i(t_{1i}) = \frac{-f_{ci}}{T_1 - T_2} \left(T_1 e^{-t_{1i}/T_1} - T_2 e^{-t_{1i}/T_2} \right) + f_{ci} = f_{ci} - \varepsilon$$
(3.45)

Similarly, the tool motion should cease at the end of deceleration section. Thus,

$$f_i(t_{1i} + t_{2i} + t_{3i}) = \frac{f_{ci}}{T_1 - T_2} \left(T_1 e^{-t_{3i}/T_2} - T_2 e^{-t_{3i}/T_2} \right) = \varepsilon$$
(3.46)

where t_{1i} , t_{2i} , t_{3i} are block acceleration, constant feedrate and deceleration time respectively. It should be noted that the value of the left hand side of equation (3.45) and (3.46) converges to f_{ci} and 0 respectively at infinite time. So, a small feedrate error ε (=0.0001 /min) needs to be introduced to get a finite value of block acceleration time t_{1i} and deceleration time t_{3i} . Having known the commanded feedrate f_{ci} for the i^{th} NC block and machine time constants T_1 and T_2 , the block acceleration time t_{1i} taken to reach the desired feedrate f_{ci} starting from rest and block deceleration time t_{3i} taken to reach rest starting from f_{ci} can be calculated by solving equation (3.45) and (3.46) respectively. However, equations (3.45) and (3.46) are non-linear is nature. Thus, numerical techniques are applied using Matlab's *fsolve* command to solve them.

Having known the block acceleration and deceleration time, tool displacement during acceleration and deceleration sections can be evaluated by integrating the feedrate profile equation.

$$l_{1i} = \int_{t=0}^{t_{1i}} f(t) = \frac{-f_{ci}}{T_1 - T_2} \left(-T_1^2 \left(e^{-\frac{t_{1i}}{T_1}} - 1 \right) + T_2^2 \left(e^{-\frac{t_{1i}}{T_2}} - 1 \right) \right) + f_{ci} t_1,$$

$$l_{3i} = \int_{t=t_{1i}+t_{2i}}^{t_{1i}+t_{2i}+t_{3i}} f(t) = \frac{f_{ci}}{T_1 - T_2} \left(-T_1^2 \left(e^{-\frac{t_{3i}}{T_1}} - 1 \right) + T_2^2 \left(e^{-\frac{t_{3i}}{T_2}} - 1 \right) \right)$$
(3.47)

Depending on the block pseudo length l_i , there can be following two cases:

Case 1: $l_i \ge l_{1i} + l_{3i}$

In this case, the block length is long enough to reach the commanded feedrate during the travel as shown in the figure (3.4). In this case the maximum feedrate attained during the travel is f_{ci} . Thus, $f_{max,i} = f_{ci}$. The distance travelled l_{2i} in constant feedrate zone and constant feedrate time t_{2i} are calculated as

$$l_{2i} = l_i - (l_{1i} + l_{3i}); \quad t_{2i} = \frac{l_{2i}}{f_{max,i}}$$
(3.48)

37

Case 2: $l_i < l_{1i} + l_{3i}$

In this case, toolpath length is not long enough to allow the tool to attain or maintain the commanded feedrate f_{ci} during the travel. Thus, $l_{2i} = t_{2i} = 0$. Furthermore, the time corresponding to the acceleration and deceleration sections, t_{1i} and t_{3i} needs to be recalculated. For this purpose, it is assumed that the maximum feedrate attained during the travel is $f_{max,i} < f_{ci}$ which is mathematically expressed as

$$f_i(t_{1i}) = \frac{-f_{ci}}{T_1 - T_2} \left(T_1 e^{-\frac{t_{1i}}{T_1}} - T_2 e^{-\frac{t_{1i}}{T_2}} \right) + f_{ci} = f_{max,i}$$
(3.49)

Reformulating equation (3.46) for this case,

$$f_i(t_{1i} + t_{2i} + t_{3i}) = \frac{f_{max,i}}{T_1 - T_2} \left(T_1 e^{-\frac{t_{3i}}{T_1}} - T_2 e^{-\frac{t_{3i}}{T_2}} \right) = \varepsilon$$
(3.50)

Furthermore, the total displacement at the end of deceleration section should be equal to block pseudo length l_i ,

$$l_{i} = \frac{-f_{ci}}{T_{1} - T_{2}} \left(-T_{1}^{2} \left(e^{-\frac{t_{1i}}{T_{1}}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{t_{1i}}{T_{2}}} - 1 \right) \right) + f_{ci} t_{1i} + \frac{f_{max,i}}{T_{1} - T_{2}} \left(-T_{1}^{2} \left(e^{-\frac{t_{3i}}{T_{1}}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{t_{3i}}{T_{2}}} - 1 \right) \right)$$
(3.51)

Equations (3.49), (3.50) and (3.51) form system of non-linear equations which are solved iteratively using trust-region non-linear optimization technique explained in the section (3.2.1) to find the unknown values of acceleration time, t_{1i} , deceleration time, t_{3i} , and the maximum feedrate attained, $f_{max,i}$. To summarize, for i^{th} block, the feedrate and displacement profiles are:

$$\left(\frac{-f_{ci}}{T_1 - T_2} \left(T_1 e^{-\frac{t}{T_1}} - T_2 e^{-\frac{t}{T_2}}\right) + f_{ci} \right), \qquad 0 \le t < t_{1i}$$

$$f_{i}(t) = \begin{cases} f_{max,i} , & t_{1i} \le t < t_{1i} + t_{2i} (3.52) \\ \frac{f_{max,i}}{T_{1} - T_{2}} \left(T_{1}e^{-\frac{(t - t_{1i} - t_{2i})}{T_{1}}} - T_{2}e^{-\frac{(t - t_{1i} - t_{2i})}{T_{2}}} \right), & t_{1i} + t_{2i} \le t \le t_{1i} + t_{2i} + t_{3i} \end{cases}$$

$$38$$

$$\begin{split} l_{i}(t) \\ &= \begin{cases} \frac{-f_{ci}}{T_{1} - T_{2}} \left(-T_{1}^{2} \left(e^{-\frac{t}{T_{1}}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{t}{T_{2}}} - 1 \right) \right) + f_{ci}t \ , \\ f_{max,i}(t - t_{1i}) + l_{i}(t_{1i}) \ , \\ f_{max,i}\left(-T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) \right) + L(t_{1i} + t_{1i}) \ , \\ f_{max,i}\left(-T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) \right) + L(t_{1i} + t_{1i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) \right) + L(t_{1i} + t_{1i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) \right) + L(t_{1i} + t_{1i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) \right) + L(t_{1i} + t_{1i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) \right) + L(t_{1i} + t_{1i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) \right) + L(t_{1i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + L(t_{1i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + T_{2}^{2} \left(e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + L(t_{1i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + L(t_{1i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{1i} - t_{2i})}{T}} - 1 \right) + L(t_{2i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{2i} - t_{2i})}{T}} - 1 \right) + L(t_{2i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{2i} - t_{2i})}{T}} - 1 \right) + L(t_{2i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{2i} - t_{2i})}{T}} - 1 \right) + L(t_{2i} + t_{2i}) \ , \\ f_{max,i}\left(e^{-T_{2i}} e^{-\frac{(t - t_{2i} - t_{2i})}{T}} - 1 \right)$$

$$\left(\frac{J_{max,i}}{T_1 - T_2} \left(-T_1^2 \left(e^{-\frac{(t-t_{11}-t_{21})}{T_1}} - 1 \right) + T_2^2 \left(e^{-\frac{(t-t_{11}-t_{21})}{T_2}} - 1 \right) \right) + l_i(t_{1i} + t_{2i}), \quad t_{1i} + t_{2i} < t \le t_{1i} + t_{2i} + t_{3i}$$

$$(3.53)$$

where i = 1, 2 ..., N. The pseudo toolpath length l_i and commanded feedrate f_{ci} data are given in the part program, machine time constants T_1 and T_2 are identified as 0.033 and 0.049 seconds respectively in section 3.2.1 and acceleration time t_{1i} , constant feedrate time t_{2i} , deceleration time t_{3i} and maximum feedrate achieved $f_{max,i}$ are calculated by equation (3.45) to (3.51). Having calculated the acceleration, constant feedrate and deceleration time corresponding to each block in NC part program, we can now proceed to evaluation of cycle time for the entire toolpath. This is based on the mode (exact-stop or continuous) selected by the user.

Exact-stop mode: In this case, the tool is allowed to decelerate to zero velocity before the execution of the next block starts as shown in the figure 3.7. Cycle time in this mode will be the sum of the acceleration time, constant feedrate time and deceleration time of all the blocks of NC code.

$$cycle \ time_{exact-stop} = \sum_{i=1}^{N} (t_{1i} + t_{2i} + t_{3i})$$
(3.52)

Continuous mode: When the machine is commanded to operate in continuous mode, the execution of the next block starts as the feedrate profile corresponding to the current block starts decelerating as shown in the figure 3.8. Thus, cycle time can be calculated by summing up the acceleration and constant feedrate time for all the blocks and the deceleration time of the last block of NC code.

$$cycle \ time_{cont.} = \sum_{i=1}^{N} (t_{1i} + t_{2i}) + t_{3N}$$
(3.53)

39



Figure 3.13 Overview of proposed cycle time prediction scheme

3.4 Experimental validation

The validation of the proposed cycle time prediction method is done in two steps. First, the experimental and predicted feedrate, acceleration and jerk profiles are compared for short toolpaths. The Mori Seiki NMV5000 CNC machine is commanded to follow the test toolpath profile shown in figure 3.14 at a commanded feedrate of 500 mm/min. The feedrate profile prediction algorithm is based on identified trajectory and corner smoothing parameters as mentioned in section 3.3. To study the effect of closed-loop feed drive system on cycle time, the feedrate profiles are computed and plotted with and without including the drive dynamics as shown in the figure 3.15.



Figure 3.14 Test 2-axis toolpath profile



Figure 3.15 Effect of closed loop feed-drive system on feedrate profile and cycle time

From the figure 3.15, it is observed that the cycle time of NC machining process is predominantly depended upon trajectory generation and corner smoothing techniques implemented on the CNC system. The closed-loop feed drive dynamics however, does not have much influence on the cycle time. Thus, the effect of closed loop feed drive dynamics is ignored for the cycle time estimation. The experimental and predicted feedrate profiles for the toolpath shown in figure 3.14 are then compared against each other when the CNC machine is operated in exact stop and continuous mode as shown in figures 3.16 and 3.17 respectively.



Figure 3.16 Experimental and predicted feedrate profiles in exact stop mode



Figure 3.17 Experimental and predicted feedrate profiles in continuous mode

Furthermore, the experimental acceleration and jerk profiles are compared against the predicted profiles. For this, the CNC machine is commanded to travel 100mm in x-direction at a feedrate of 6000 mm/min. The measured and predicted trajectory profiles are shown in figure 3.18. From the plotted graph, we can conclude that the predicted and actual kinematic profiles are in good match with each other.



Figure 3.18 Experimental and predicted trajectory profiles

In the next step, the cycle time estimated by the proposed algorithm is compared with the measured cycle time for machining aerospace components in both exact-stop and continuous mode as shown in table 3.3. The first test model is corresponding to 3-axis toolpath provided by the Boeing Company. The error in cycle time prediction for this model is 2.3% and 4.5% in exact stop and continuous mode respectively. The second test model which involves 5-axis toolpath, has cycle time prediction error of 1.7% and 3.33% in the exact stop and continuous mode respectively. The comparatively higher prediction error in continuous mode is due to added error in modeling of corner feedrate.

	Test models		
		(3-Axis Toolpath)	(5-axis Toolpath)
	Measured cycle time (sec)	16217	2001
Exact- stop mode	Predicted cycle time (sec)	15842	1967
	Prediction error (%)	2.3	1.7
	Measured cycle time (sec)	3372	290
Continuous mode	Predicted cycle time (sec)	3526	300
	Prediction error (%)	4.5	3.33

Table 3.3 Experimental and predicted cycle time

3.5 Summary

Estimation of cycle time for milling process is important for process planning and analysis of machine tool capabilities. In this chapter, we proposed a scheme to predict cycle time for milling free-form surfaces based on identified CNC machine dynamics in exact-stop and continuous mode. For this, simple sets of experiments have been conducted in Mori Seiki NMV5000 DCG commercial CNC machine to identify the implemented trajectory generation and corner smoothing technique and feed drive dynamics of the CNC system. From experimental data, it is concluded 45

that the cycle time of NC machining process is predominantly effected by trajectory generation and corner smoothing techniques implemented on the CNC system. However, the closed-loop feed drive dynamics does not have much influence on the cycle time. An analytical approach for predicting cycle time based on the identified CNC system dynamics and given part program is presented. To validate the proposed cycle time prediction algorithm, the predicted feedrate profile and cycle time are compared with the experimentally measured data. Experimental results show that the cycle time prediction error remains within 5% for various 2-axis, 3-axis and 5-axis toolpaths in both exact-stop and continuous mode.

Chapter 4: Five-axis corner smoothing for CNC system

4.1 Overview

Five-axis machine tools are widely used for manufacturing of sculptured surfaces in aerospace, die and mold industries. In most of the cases, motion commands are composed of a series of linear toolpath segments, also called as G01 commands. If the tool motion is ceased at all the block transitions, NC cycle time increases and productivity suffers. On the other hand, if the tool is commanded to maintain its speed across the sharp corners, discontinuities are created in velocity, acceleration and jerk commands due to geometrical discontinuities in the toolpath. This will result in high impact load on the feed drives causing machine tool vibrations and can damage the machined surface.

In this chapter, a novel decoupled approach as shown in figure 4.2 is presented to achieve local corner smoothing of five-axis toolpaths. The flow chart of trajectory generation for five-axis corner smoothing is shown in figure 4.1. Toolpath position and orientation are smoothed by inserting quintic and normalized septic micro-splines, respectively between the adjacent linear toolpath segments. Optimal control points are calculated for position splines to achieve C^3 continuity at the junctions between the splines and the linear segments while respecting user-defined corner position tolerance (ε_{pos}) limits in Section 4.2. Likewise, Section 4.3 focusses on constraint equations formulation based on pre-defined orientation tolerance (ε_{ori}) limits, continuity at junctions and synchronization of position and orientation splines for calculation of optimal control points for normalized septic orientation spline. After geometrical modification of the toolpath, feedrate planning is performed using cubic acceleration feedrate profile in section 4.4. The linear and the spline sections of the toolpath are interpolated by linear and spline interpolation techniques respectively. The non-linear relation between spline parameter u and displacement along the toolpath s is approximated using ninth-order feed correction spline. Inverse kinematics is then performed on the interpolated position and orientation data to generate axis motion commands. In case drive acceleration or jerk limits are violated, commanded feedrate is adjusted using the scheme proposed by Sencer et. al. [41] and position and orientation data are interpolated again. In Section 4.5, the smoothed toolpath motion is compared against the unsmooth motion in experiments and in simulation to show improvements in reducing the cycle time while improving tracking accuracy in five-axis machining of free-form surfaces found in dies, molds and aerospace parts.

4.2 Toolpath position smoothing

As shown in figure 4.2, five-axis corner is defined by two adjoining linear toolpath segments with tool tip position and orientation vectors represented as $\mathbf{p}_i = [x_i, y_i, z_i]$ and $\mathbf{o}_i = [o_{ii}, o_{ji}, o_{ki}]$ respectively for tool position i = 1,2,3 such that $\|\mathbf{p}_1 - \mathbf{p}_2\| = l_1, \|\mathbf{p}_2 - \mathbf{p}_3\| = l_2, \mathbf{o}_1 \cdot \mathbf{o}_2 = \cos \theta_1$ and $\mathbf{o}_2 \cdot \mathbf{o}_3 = \cos \theta_2$. Various works are present in literature addressing toolpath position smoothing as mentioned in chapter 2. However, none of the methods could achieve C^3 continuous toolpath. Thus, a new toolpath position smoothing technique is proposed. The sharp position corners are smoothed by fitting quintic micro-splines between linear toolpath segments at each block transition point. In this section, formulation of quintic B-spline is presented. Optimal control points for the micro-splines are calculated based on pre-defined position error tolerance limit ε_{pos} and conditions for acceleration and jerk continuity at the junction between the linear toolpath segments.

4.2.1 Parametric position micro-spline

A parametric position micro-spline which is inserted between the two linear toolpath segments is defined by the basis functions $N_{i,k}(u)$, (n + 1) control points $P_i = [P_{xi}, P_{yi}, P_{zi}]$ and degree (k - 1) with the following form:

$$\boldsymbol{P}(u) = \sum_{i=0}^{n} N_{i,k}(u) \boldsymbol{P}_{i}$$
(4.1)



Figure 4.1 Trajectory generation for five-axis corner smoothing



Figure 4.2 Decoupled method of five-axis corner smoothing

The basis functions $N_{i,k}(u)$ are functions of the geometric parameter u and knot vector $U = [u_0, u_1, ..., u_{n+k}]$ and are defined as follows:

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

Where 0/0 = 0, and with the limiting/initial condition (for k = 1) of:

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } u_i \le u \le u_{i+1} \\ 0 & \text{otherwise} \end{cases} ; \quad u \in [0,1]$$

$$(4.2)$$

The degree of the spline curve and number of control points are set as 5 and 7 respectively to ensure continuity in third geometric derivative. The non-periodic knot vector is constructed as $[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1]$ to ensure symmetry across the angular bisector of the corner angle. For 5th degree spline curve with 7 control points, the spline equation (4.1) can be expressed as

$$P(u) = \sum_{i=0}^{6} N_{i,6}(u) P_i$$
(4.3)

where the values of basis functions $N_{i,6}$ for i = 0,1,...,6 are evaluated using equation (4.2) and plotted as shown in figure 4.3.

$$N_{0,6}(u) = \begin{cases} -32u^5 + 80u^4 - 80u^3 + 40u^2 - 10u + 1, & u < 0.5\\ 0, & 0.5 \le u \le 1 \end{cases};$$

$$N_{1,6}(u) = \begin{cases} 62u^5 - 150u^4 + 140u^3 - 60u^2 + 10u, & u < 0.5\\ -2u^5 + 10u^4 - 20u^3 + 20u^2 - 10u + 2, & 0.5 \le u \le 1 \end{cases};$$

$$N_{2,6}(u) = \begin{cases} -52u^5 + 110u^4 - 80u^3 + 20u^2, & u < 0.5\\ 12u^5 - 50u^4 + 80u^3 - 60u^2 + 20u - 2, & 0.5 \le u \le 1 \end{cases};$$

$$N_{3,6}(u) = \begin{cases} 32u^5 - 50u^4 + 20u^3, & u < 0.5 \\ -32u^5 + 110u^4 - 140u^3 + 80u^2 - 20u + 2, & 0.5 \le u \le 1 \end{cases};$$

$$N_{4,6}(u) = \begin{cases} -12u^5 + 10u^4, & u < 0.5\\ 52u^5 - 150u^4 + 160u^3 - 80u^2 + 20u - 2, & 0.5 \le u \le 1 \end{cases};$$

$$N_{5,6}(u) = \begin{cases} 2u^5, & u < 0.5\\ -62u^5 + 160u^4 - 160u^3 + 80u^2 - 20u + 2, & 0.5 \le u \le 1 \end{cases};$$

$$N_{6,6}(u) = \begin{cases} 0, & u < 0.5\\ 32u^5 - 80u^4 + 80u^3 - 40u^2 + 10u - 1, & 0.5 \le u \le 1 \end{cases}$$



Figure 4.3 Plot of basis function corresponding to different control points for quintic B-Spline

On substituting the values of basis functions $N_{i,6}$ for i = 0,1, ..., 6 in the equation (4.3), the position micro-spline equation is obtained as

$$P(u) = \begin{cases} \begin{bmatrix} -32u^{5} + 80u^{4} - 80u^{3} + 40u^{2} - 10u + 1\\ 62u^{5} - 150u^{4} + 140u^{3} - 60u^{2} + 10u\\ -52u^{5} + 110u^{4} - 80u^{3} + 20u^{2}\\ 32u^{5} - 50u^{4} + 20u^{3}\\ -12u^{5} + 10u^{4}\\ 2u^{5}\\ 0 \end{bmatrix} \begin{bmatrix} P_{0}\\ P_{1}\\ P_{2}\\ P_{3}\\ P_{4}\\ P_{5}\\ P_{6} \end{bmatrix}^{T}, \quad u < 0.5 \end{cases}$$

$$P(u) = \begin{cases} \begin{bmatrix} -2u^{5} + 10u^{4} - 20u^{3} + 20u^{2} - 10u + 2\\ 12u^{5} - 50u^{4} + 80u^{3} - 60u^{2} + 20u - 2\\ -32u^{5} + 110u^{4} - 140u^{3} + 80u^{2} - 20u + 2\\ 52u^{5} - 150u^{4} + 160u^{3} - 80u^{2} + 20u - 2\\ -62u^{5} + 160u^{4} - 160u^{3} + 80u^{2} - 20u + 2\\ 32u^{5} - 80u^{4} + 80u^{3} - 40u^{2} + 10u - 1 \end{bmatrix} \begin{bmatrix} P_{0}\\ P_{1}\\ P_{2}\\ P_{3}\\ P_{4}\\ P_{5}\\ P_{6} \end{bmatrix}^{T}, \quad 0.5 \le u \le 1 \end{cases}$$

$$(4.4)$$

However, the positions of the control points $P_i \forall i = 0, 1, 2, ..., 6$ are yet to be calculated to completely define the position micro-spline P(u).

4.2.2 Optimal control point calculation for position micro-spline

The choice of non-periodic knot vector ensures that the B-spline curve will pass through the first and the last control point P_0 ($P(u = 0) = P_0$) and P_6 ($P(u = 1) = P_6$), respectively, and tangential to the first and the last segment of the control polygon P_0P_1 and P_5P_6 , respectively as shown in figure 4.4. In order to maintain the position and tangential continuity at the junctions between the linear segments and the position micro-splines, the first three control points P_0, P_1 and P_2 should be located in the toolpath segment p_1p_2 . Likewise, the last three control points P_4, P_5 and P_6 should be located in the toolpath segment p_2p_3 . The fourth control point P_3 is chosen at the corner position $p_2 = [x_2, y_2, z_2]$ to ensure symmetrical spline curve. However, constraints based on position error tolerance limit ε_{pos} and conditions for acceleration and jerk continuity at junctions P_0 (u = 0) and $P_6(u = 1)$ need to be imposed to find the exact locations of the other six control points.



Figure 4.4 Toolpath position smoothing using quintic B-spline

4.2.2.1 Acceleration and jerk continuity at junctions

In order to achieve acceleration and jerk continuity at the junction points P_0 (u = 0) and $P_6(u = 1)$, C^3 continuity of the toolpath at the junction points are needed to be ensured first as shown in figure 4.5. The second and the third derivative of the position vector P with respect to the toolpath length *s* for a linear toolpath profile is always zero, i.e.,

$$\frac{d^2 \boldsymbol{P}}{ds^2} = \begin{bmatrix} 0\\0\\0 \end{bmatrix}^T; \quad \frac{d^3 \boldsymbol{P}}{ds^3} = \begin{bmatrix} 0\\0\\0 \end{bmatrix}^T$$
(4.5)



Figure 4.5 Constraints for acceleration and jerk continuities at junctions

Differentiating toolpath position vector P(u) to evaluate the second and third derivatives of the quintic spline profile,

$$\frac{d^2 \mathbf{P}}{ds^2} = \frac{d^2 \mathbf{P}}{du^2} \left(\frac{du}{ds}\right)^2 + \frac{d\mathbf{P}}{du} \frac{d^2 u}{ds^2}; \qquad (4.6)$$

$$\frac{d^3 \mathbf{P}}{ds^3} = \frac{d^3 \mathbf{P}}{du^3} \left(\frac{du}{ds}\right)^3 + 3\frac{d^2 \mathbf{P}}{du^2}\frac{du}{ds}\frac{d^2 u}{ds^2} + \frac{d\mathbf{P}}{du}\frac{d^3 u}{ds^3}$$
(4.7)

From equations (4.5), (4.6) and (4.7), it can be concluded that the C^3 continuity will be maintained at the junctions if

$$\left(\frac{d^2 \mathbf{P}}{du^2} \left(\frac{du}{ds}\right)^2 + \frac{d\mathbf{P}}{du} \left.\frac{d^2 u}{ds^2}\right)\right|_{u=0,1} = \begin{bmatrix} 0\\0\\0\end{bmatrix}^T, and$$
(4.8)

$$\left(\frac{d^3\boldsymbol{P}}{du^3}\left(\frac{du}{ds}\right)^3 + 3\frac{d^2\boldsymbol{P}}{du^2}\frac{du}{ds}\frac{d^2u}{ds^2} + \frac{d\boldsymbol{P}}{du}\frac{d^3u}{ds^3}\right)\Big|_{u=0,1} = \begin{bmatrix}0\\0\\0\end{bmatrix}^T$$
(4.9)

The sufficient conditions for (4.8) and (4.9) to be true are

$$\frac{d^{2}\boldsymbol{P}}{du^{2}}\Big|_{u=0,1} = \begin{bmatrix} 0\\0\\0 \end{bmatrix}^{T}; \frac{d^{3}\boldsymbol{P}}{du^{3}}\Big|_{u=0,1} = \begin{bmatrix} 0\\0\\0 \end{bmatrix}^{T} and,$$
(4.10)

$$\frac{d^2 u}{ds^2}\Big|_{u=0,1} = 0; \frac{d^3 u}{ds^3}\Big|_{u=0,1} = 0$$
(4.11)

Now, the first derivative of the spline parameter u with respect to the path length s can be written as:

$$\frac{du}{ds} = \frac{1}{\left(\frac{ds}{du}\right)} = \frac{1}{\sqrt{\left(\frac{dP_x}{du}\right)^2 + \left(\frac{dP_y}{du}\right)^2 + \left(\frac{dP_z}{du}\right)^2}} = \frac{1}{f(u)}$$
(4.12)

where $ds = \sqrt{(dP_x)^2 + (dP_y)^2 + (dP_z)^2}$ and $P(u) = P_x \hat{i} + P_y \hat{j} + P_z \hat{k}$. Differentiating equation (4.12) with respect to the toolpath length *s* to evaluate the second and third derivatives of the spline parameter *u*,

$$\frac{d^{2}u}{ds^{2}} = -\frac{f'(u)}{(f(u))^{2}} \frac{du}{ds} = -\frac{f'(u)}{(f(u))^{3}}$$
$$= \frac{\left(\frac{dP_{x}}{du} \frac{d^{2}P_{x}}{du^{2}} + \frac{dP_{y}}{du} \frac{d^{2}P_{y}}{du^{2}} + \frac{dP_{z}}{du} \frac{d^{2}P_{z}}{du^{2}}\right)}{\left(\left(\frac{dP_{x}}{du}\right)^{2} + \left(\frac{dP_{y}}{du}\right)^{2} + \left(\frac{dP_{z}}{du}\right)^{2}\right)^{2}} = \frac{\left(\frac{dP}{du}\right)\left(\frac{d^{2}P}{du^{2}}\right)^{T}}{\left\|\frac{dP}{du}\right\|^{4}}$$
(4.13)

$$\frac{d^{3}u}{ds^{3}} = \frac{3(f'(u))^{2} - f(u)f''(u)}{(f(u))^{4}} \frac{du}{ds} = \frac{3(f'(u))^{2} - f(u)f''(u)}{(f(u))^{5}} \\
= \frac{3\left(\left(\frac{dP}{du}\right)\left(\frac{d^{2}P}{du^{2}}\right)^{T}\right)^{2} - \left\|\frac{dP}{du}\right\|^{2}\left(\left(\frac{d^{2}P}{du^{2}}\right)\left(\frac{d^{2}P}{du^{2}}\right)^{T} + \left(\frac{dP}{du}\right)\left(\frac{d^{3}P}{du^{3}}\right)^{T}\right) - \left\|\frac{dP}{du}\right\|\left(\left(\frac{dP}{du}\right)\left(\frac{d^{2}P}{du^{2}}\right)^{T}\right) \\
= \frac{\left\|\frac{dP}{du}\right\|^{7}}{\left\|\frac{dP}{du}\right\|^{7}} \tag{4.14}$$

56

From equation (4.13) and (4.14), it can be observed that the values of (d^2u/ds^2) and (d^3u/ds^3) ceases as the values of $(d^2\mathbf{P}/du^2)$ and $(d^3\mathbf{P}/du^3)$ goes to zero.

$$\frac{d^2 \mathbf{P}}{du^2} = \begin{bmatrix} 0\\0\\0 \end{bmatrix}^T, \frac{d^3 \mathbf{P}}{du^3} = \begin{bmatrix} 0\\0\\0 \end{bmatrix}^T \to \frac{d^2 u}{ds^2} = 0, \frac{d^3 u}{ds^3} = 0$$
(4.15)

Thus, equation (4.10) is the sufficient criteria for equation (4.11) to be true which implies C^3 continuity can be achieved at the junctions if only equation (4.10) is satisfied. From equation (4.4), the second derivative of the position vector **P** with respect to the spline parameter *u* can be expressed as

$$\frac{d^{2}P}{du^{2}} = \begin{cases} \begin{bmatrix} -640u^{3} + 960u^{2} - 480u + 80\\ 1240u^{3} - 1800u^{2} + 840u - 120\\ -1040u^{3} + 1320u^{2} - 480u + 40\\ 640u^{3} - 600u^{2} + 120u\\ -240u^{3} + 120u^{2}\\ 40u^{3}\\ 0 \end{bmatrix} \begin{bmatrix} P_{0}\\ P_{1}\\ P_{2}\\ P_{3}\\ P_{4}\\ P_{5}\\ P_{6} \end{bmatrix}^{T}, \quad u < 0.5 \end{cases}$$

$$\begin{pmatrix} \frac{d^{2}P}{du^{2}} = \begin{cases} 0\\ -40u^{3} + 120u^{2} - 120u + 40\\ 240u^{3} - 600u^{2} + 480u - 120\\ -640u^{3} + 1320u^{2} - 840u + 160\\ 1040u^{3} - 1800u^{2} + 960u - 160\\ -1240u^{3} + 1920u^{2} - 960u + 160\\ 640u^{3} - 960u^{2} + 480u - 80 \end{bmatrix} \begin{bmatrix} P_{0}\\ P_{1}\\ P_{2}\\ P_{3}\\ P_{4}\\ P_{5}\\ P_{6} \end{bmatrix}, \quad 0.5 \le u \le 1 \end{cases}$$

$$(4.16)$$

Differentiating (4.16) to evaluate the third derivative of the position vector P with respect to the spline parameter u, we get

Substituting u = 0, 1 in (4.16) to evaluate the values of $(d^2 P/du^2)$ at the junctions P_0 and P_1 respectively, we get

$$\frac{d^{2}\boldsymbol{P}}{du^{2}}\Big|_{u=0} = \begin{bmatrix} 80\\ -120\\ 40\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{0}\\ \boldsymbol{P}_{1}\\ \boldsymbol{P}_{2}\\ \boldsymbol{P}_{3}\\ \boldsymbol{P}_{4}\\ \boldsymbol{P}_{5}\\ \boldsymbol{P}_{6}\end{bmatrix}^{T}; \frac{d^{2}\boldsymbol{P}}{du^{2}}\Big|_{u=1} = \begin{bmatrix} 0\\ 0\\ 0\\ 0\\ 40\\ -120\\ 80 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{0}\\ \boldsymbol{P}_{1}\\ \boldsymbol{P}_{2}\\ \boldsymbol{P}_{3}\\ \boldsymbol{P}_{4}\\ \boldsymbol{P}_{5}\\ \boldsymbol{P}_{6}\end{bmatrix}^{T}$$
(4.18)

Substituting u = 0, 1 in (4.17) to evaluate the values of $(d^3 P/du^3)$ at the junctions P_0 and P_1 respectively, we get

$$\frac{d^{3}\boldsymbol{P}}{du^{3}}\Big|_{u=0} = \begin{bmatrix} -480\\ 840\\ -480\\ 120\\ 0\\ 0\\ 0\\ 0\\ 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{0}\\ \boldsymbol{P}_{1}\\ \boldsymbol{P}_{2}\\ \boldsymbol{P}_{3}\\ \boldsymbol{P}_{4}\\ \boldsymbol{P}_{5}\\ \boldsymbol{P}_{6}\end{bmatrix}^{T}; \frac{d^{3}\boldsymbol{P}}{du^{3}}\Big|_{u=1} = \begin{bmatrix} 0\\ 0\\ 0\\ -120\\ 480\\ -840\\ 480 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{0}\\ \boldsymbol{P}_{1}\\ \boldsymbol{P}_{2}\\ \boldsymbol{P}_{3}\\ \boldsymbol{P}_{4}\\ \boldsymbol{P}_{5}\\ \boldsymbol{P}_{6}\end{bmatrix}^{T}$$
(4.19)

From equation (4.18) and (4.19), the condition for C^3 continuity at the junction $P_0(u = 0)$ is found as

$$80P_0 - 120P_1 + 40P_2 = 0,$$

-480P_0 + 840P_1 - 480P_2 + 120P_3 = 0 (4.20)

On solving (4.20), we get:

$$P_{1} = 2P_{2} - P_{3} \quad and$$

$$P_{0} = \frac{5P_{2} - P_{3}}{2}$$
(4.21)

Similarly, the condition for C^3 continuity at the junction $P_6(u = 1)$ is found as

$$80\mathbf{P}_6 - 120\mathbf{P}_5 + 40\mathbf{P}_4 = 0,$$

$$480\mathbf{P}_6 - 840\mathbf{P}_5 + 480\mathbf{P}_4 - 120\mathbf{P}_3 = 0$$
 (4.22)

On solving (4.22), we get:

$$P_{5} = 2P_{4} - P_{3} \quad and$$

$$P_{6} = \frac{5P_{4} - P_{3}}{2} \tag{4.23}$$

As shown in figure 4.6, the length of the line segment P_2P_3 and P_3P_4 are kept same (*l*) for symmetry purpose. From equations (4.21) and (4.23), the lengths of the line segments P_0P_3 , P_1P_3 , P_3P_5 and P_3P_6 are evaluated as 2.5*l*, 2*l*, 2*l* and 2.5*l* respectively. It should be noted here that one more condition is required to evaluate the value of *l* to find the exact locations of the position control points. This is done by formulating user-defined position error tolerance limit constraint as discussed in the following section.

4.2.2.2 Pre-defined position error tolerance limit, ε_{pos}

The B-spline curve defined by equation 4.4 is formulated that the maximum position error e_{max} occurs corresponding to the mid-point of the spline. By substituting u = 0.5 in equation 4.4, we

get,

$$\boldsymbol{P}(0.5) = \begin{bmatrix} 0\\1/16\\1/4\\3/8\\1/4\\1/16\\0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{0}\\\boldsymbol{P}_{1}\\\boldsymbol{P}_{2}\\\boldsymbol{P}_{3}\\\boldsymbol{P}_{4}\\\boldsymbol{P}_{5}\\\boldsymbol{P}_{6} \end{bmatrix}^{T}$$
(4.24)

The maximum position error is evaluated as

$$e_{max} = \mathbf{P}_{3} - \mathbf{P}(0.5)$$

$$= \mathbf{P}_{3} - \left(\frac{1}{16}\mathbf{P}_{1} + \frac{1}{4}\mathbf{P}_{2} + \frac{3}{8}\mathbf{P}_{3} + \frac{1}{4}\mathbf{P}_{4} + \frac{1}{16}\mathbf{P}_{5}\right)$$

$$= \frac{5}{8}\mathbf{P}_{3} - \frac{1}{4}\mathbf{P}_{4} - \frac{1}{4}\mathbf{P}_{2} - \frac{1}{16}\mathbf{P}_{5} - \frac{1}{16}\mathbf{P}_{1}$$

$$= \frac{1}{4}(\mathbf{P}_{3} - \mathbf{P}_{4}) + \frac{1}{4}(\mathbf{P}_{3} - \mathbf{P}_{2}) + \frac{1}{16}(\mathbf{P}_{3} - \mathbf{P}_{5}) + \frac{1}{16}(\mathbf{P}_{3} - \mathbf{P}_{1})$$
(4.25)

From section 4.2.2.1, the lengths of the line segments P_1P_3 , P_2P_3 , P_3P_4 and P_3P_5 are evaluated as 2*l*, *l*, *l* and 2*l* respectively. Using this and squaring both the sides of the equation (4.25), we get,

$$e_{max}^{2} = 2\left(2\left(\frac{l}{4}\right)^{2} + \left(\frac{2l}{16}\right)^{2} + 2\left(\frac{l}{4}\right)^{2}\cos\alpha + \left(\frac{2l}{16}\right)^{2}\cos\alpha\right)$$
$$= \frac{9}{32}l^{2}(1 + \cos\alpha)$$
$$= \frac{9}{16}l^{2}\cos\left(\frac{\alpha}{2}\right)$$
(4.26)

where α is the corner angle. As the maximum position error e_{max} incurred during the corner smoothing of the toolpath should be less than the user-defined position error tolerance limit ε_{pos} ,

$$\|e_{max}\| = \frac{4}{3}l\cos\left(\frac{\alpha}{2}\right) \le \varepsilon_{pos}$$

$$Or, l \le \left(\frac{4\varepsilon_{pos}}{3\cos\left(\frac{\alpha}{2}\right)}\right)$$
(4.27)

As each linear toolpath segment is used to smooth two corner points corresponding to the start and end-point of the segment, the maximum allowable length of the line segment connecting the control points is half the length of the toolpath segment. Mathematically, it can be expressed as,

$$2.5l \le 0.5l_1,$$

 $2.5l \le 0.5l_2$ (4.28)

Combining (4.27) and (4.28),



Figure 4.6 Location of position control points based on continuity and error tolerance constraints
4.3 Toolpath orientation smoothing

In comparison to tool tip position smoothing, tool orientation smoothing is mathematically more challenging. X. Buedaert et al [27] carried out local smoothing of toolpath orientation within given orientation tolerance by using dual-spline method. To obtain a smooth connection of the orientation a parameterization spline was employed to link the bottom and top B-Spline parameters. However, due to over-constraining the problem, the proposed solution is highly sensitive to even a tiny variation in orientation. Moreover, dual-spline method does not work well when there is a large orientation change for a very small position change. Because of the limitations of dual-spline method as an orientation interpolation method, Yuen et al [18] interpolated position and orientation separately. Unlike interpolation of tool tip position data, the tool orientation was not fitted directly to the given orientation data. Instead, to ensure that the norm of orientation vectors always remain unity, the tool orientations were mapped to the spherical co-ordinates and orientation spline was then fitted to spherical co-ordinate data. This method worked effectively for global smoothing of tool orientations. However, for local smoothing of tool orientations, interpolation needs to be carried out directly from the orientation data in work-piece co-ordinate frame and should not be converted to spherical co-ordinate. This is because it is difficult to formulate user defined orientation tolerance constraints in the spherical co-ordinates.

Linear quaternion interpolation (*lerp*) is commonly used for linear interpolation between rotation quaternion [42]. Given two quaternions q_0, q_1 and $h \in [0,1]$ this interpolation curve can be stated: $lerp(q_0, q_1, h) = q_0(1 - h) + q_1h$ (4.30)

The interpolation curve for linear interpolation between quaternions gives a straight line in the quaternion space. The curve therefore dips below the surface of the unit sphere. Since all quaternions on a line through the origin gives the same rotation, the curve can be projected on to the unit sphere by normalizing equation (4.30) without changing the corresponding rotations. This idea is implemented to the standard spline interpolation technique to formulate a novel normalized spline interpolation method which satisfies both continuity and unity norm criteria.

The tool orientation vectors are represented as $o_i = [o_{ii}, o_{ji}, o_{ki}]$ for tool position i = 1,2,3 and it lies in the surface of unity sphere. The angle between o_1 and o_2 is assumed as θ_1 and between o_2 and o_3 is assumed as θ_2 . Mathematically, this can be expressed as

$$\theta_1 = \cos^{-1}(\boldsymbol{o}_1 \cdot \boldsymbol{o}_2); \ \theta_2 = \cos^{-1}(\boldsymbol{o}_2 \cdot \boldsymbol{o}_3) \tag{4.31}$$



Figure 4.7 Toolpath orientation smoothing using Normalized Heptic B-Spline

As shown in figure 4.7, tool orientation is smoothed by inserting **normalized heptic** B-spline with 9 orientation control points between linearly interpolated orientation vectors unlike quintic position B-spline with 7 control points for position smoothing. The order of inserted spline segments for orientation smoothing is increased due to extra constraints based on synchronization of position and orientation spline in this case.

4.3.1 Parametric orientation micro-spline

The 7th degree heptic spline curve with given control points $\boldsymbol{O}_i = [O_{ii}, O_{ji}, O_{ki}] \forall i = 0, 1, ..., 8$ and non-periodic knot vector $U = [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]$ is constructed using (4.1) and (4.2) as

 $\boldsymbol{B}(u)$

Since length of the tool cannot change, the magnitude of orientation vector must be unity at all times to ensure feasible tool orientations. To ensure this, the heptic spline is divided by ||B(u)|| to obtain normalised orientation spline O(u).

$$\boldsymbol{O}(u) = \frac{\boldsymbol{B}(u)}{\|\boldsymbol{B}(u)\|}, \quad u \in [0,1]$$
(4.33)

Thus the normalized orientation heptic B-spline can be represented as

n

$$\boldsymbol{O}(u) = \sum_{i=0}^{n} (N_{norm})_{i,k}(u) \boldsymbol{O}_i$$
(4.34)

where the basis function for the normalized spline can be defined as

$$(N_{norm})_{i,k} = \frac{N_{i,k}}{\|\boldsymbol{B}(u)\|}$$
(4.35)

The basis functions for normalized and general heptic B-spline are compared against each other in figure 4.8 as shown below:



Figure 4.8 Basis functions of Heptic and Normalized Heptic B-Spline

As the norms of all the orientation control vectors $O_i \forall i = 0, 1, ..., 8$ are unity, the value of ||B(u)|| is also very close to unity. Due to this, the basis functions of normalised heptic B-Spline almost coincides with that of heptic B-Spline as shown in the figure 4.8. Mathematically, it can be observed that the normalization term does not affect the continuity of the spline. Thus, both continuity and unity norm constraints can be satisfied using normalized spline. However, the positions of the control points O_i are yet to be calculated to completely define the orientation micro-spline.

4.3.2 Optimal control point calculation for orientation micro-spline

The orientation control points O_0 , O_1 , O_2 and O_3 which make angles δ_1 , γ_1 , β_1 and α_1 respectively with the corner orientation vector o_2 should all lie in the plane created by the orientation vectors o_1 and o_2 to respect the continuity constraint. The fifth control point O_4 is chosen as the corner orientation $o_2 = [o_{i2}, o_{j2}, o_{k2}]$ without losing generality. The control points O_5 , O_6 , O_7 and O_8 are assumed to be positioned at angles α_2 , β_2 , γ_2 and δ_2 respectively from the corner orientation vector o_2 in the plane created by the orientation vectors o_2 and o_3 as shown in figure 4.9.



Figure 4.9 Constraints for velocity, acceleration and jerk continuities at junctions for orientation spline

However, constraints based on orientation error tolerance limit ε_{ori} , synchronisation of position and orientation spline and conditions for velocity, acceleration and jerk continuity at junctions O_0 (u = 0) and $O_8(u = 1)$ need to be imposed to evaluate the value of the angles where the control points are located. It should be noted here that as the same spline parameter u is used for generation of position and orientation spline, both splines should be synchronized properly.

4.3.2.1 Velocity, acceleration and jerk continuity at junctions

In order to achieve velocity, acceleration and jerk continuity at the junction points O_0 (u = 0) and $O_8(u = 1)$, C^3 continuity of the toolpath at the junction points are needed to be satisfied first. In the linear interpolation of toolpath between the cutter location (CL) data [p_1 , o_1] and [p_2 , o_2], any intermediate orientation O which makes an angle θ with the vector o_1 is given by Rodrigues' rotation formula [43] which gives:

$$\boldsymbol{0} = \boldsymbol{o}_1 \cos \theta + (\boldsymbol{k} \times \boldsymbol{o}_1) \sin \theta + \boldsymbol{k} (\boldsymbol{k} \cdot \boldsymbol{o}_1) (1 - \cos \theta)$$
(4.36)

where \boldsymbol{k} is the unit vector defining the axis of the rotation.

$$k = \frac{o_1 \times o_2}{\|o_2 \times o_1\|} = \frac{o_1 \times o_2}{\|o_1\| \|o_2\| \sin \theta_1} = \frac{o_1 \times o_2}{\sin \theta_1} \quad (\because \|o_1\| = \|o_2\| = 1)$$
(4.37)

Substituting the value of \boldsymbol{k} in (4.36), we can write

$$\boldsymbol{0} = \boldsymbol{o}_1 \cos \theta + \left(\frac{(\boldsymbol{o}_1 \times \boldsymbol{o}_2) \times \boldsymbol{o}_1}{\sin \theta_1}\right) \sin \theta + \frac{\boldsymbol{o}_1 \times \boldsymbol{o}_2}{\sin \theta_1} \left(\frac{(\boldsymbol{o}_1 \times \boldsymbol{o}_2) \cdot \boldsymbol{o}_1}{\sin \theta_1}\right) (1 - \cos \theta)$$
(4.38)

Using properties of scalar and vector triple products, we get

$$(o_1 \times o_2) \times o_1 = -(o_1 \cdot o_2)o_1 + (o_1 \cdot o_1)o_2 = -o_1 \cos \theta_1 + o_2$$

 $(o_1 \times o_2) \cdot o_1 = 0$ (4.39)

Using (4.38) and (4.39),

$$\boldsymbol{0} = \boldsymbol{o_1} \cos \theta + \left(\frac{-\boldsymbol{o_1} \cos \theta_1 + \boldsymbol{o_2}}{\sin \theta_1}\right) \sin \theta = \frac{\boldsymbol{o_1} \cos \theta \sin \theta_1 - \boldsymbol{o_1} \cos \theta_1 \sin \theta + \boldsymbol{o_2} \sin \theta}{\sin \theta_1}$$
$$= \frac{\boldsymbol{o_1} \sin(\theta_1 - \theta) + \boldsymbol{o_2} \sin \theta}{\sin \theta_1}, \quad \theta \in [0, \theta_1]$$
(4.40)

It should also be noted that the ratio of the intermediate angle θ to the total angle θ_1 between o_1 and o_2 should be in accord with the ratio of intermediate path length s to the total path length l_1 between p_1 and p_2 :

$$\theta = \frac{s}{l_1} \theta_1 \tag{4.41}$$

Using (4.40) and (4.41), the derivative of the orientation vector \boldsymbol{O} with respect to the toolpath length *s* for a linear toolpath profile is expressed as follows:

$$\frac{d\boldsymbol{0}}{ds} = \frac{d\boldsymbol{0}}{d\theta}\frac{d\theta}{ds} = \frac{-\boldsymbol{o_1}\cos(\theta_1 - \theta) + \boldsymbol{o_2}\cos\theta}{\sin\theta_1}\left(\frac{d\theta}{ds}\right) = \frac{\theta_1}{l_1}\left(\frac{-\boldsymbol{o_1}\cos(\theta_1 - \theta) + \boldsymbol{o_2}\cos\theta}{\sin\theta_1}\right)$$
(4.42)

At the junction point
$$\theta = (\theta_1 - \delta_1)^-$$
,

$$\frac{d\boldsymbol{0}}{ds}\Big|_{\theta = (\theta_1 - \delta_1)^-} = \frac{\theta_1}{l_1} \left(\frac{-\boldsymbol{o}_1 \cos \delta_1 + \boldsymbol{o}_2 \cos(\theta_1 - \delta_1)}{\sin \theta_1}\right)$$
(4.43)

Similarly, differentiating $(d\boldsymbol{0}/ds)$ with respect to s and substituting $\theta = (\theta_1 - \delta_1)^-$, we get

$$\frac{d^2 \boldsymbol{0}}{ds^2}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_1-\boldsymbol{\delta}_1)^-} = -\left(\frac{\boldsymbol{\theta}_1}{l_1}\right)^2 \left(\frac{\boldsymbol{o}_1 \sin \boldsymbol{\delta}_1 + \boldsymbol{o}_2 \sin(\boldsymbol{\theta}_1-\boldsymbol{\delta}_1)}{\sin \boldsymbol{\theta}_1}\right) = -\left(\frac{\boldsymbol{\theta}_1}{l_1}\right)^2 \boldsymbol{0}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_1-\boldsymbol{\delta}_1)^-}$$
(4.44)

$$\frac{d^{3}\boldsymbol{\textit{0}}}{ds^{3}}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_{1}-\boldsymbol{\delta}_{1})^{-}} = -\left(\frac{\boldsymbol{\theta}_{1}}{l_{1}}\right)^{3}\left(\frac{-\boldsymbol{\textit{o}}_{1}\cos\boldsymbol{\delta}_{1}+\boldsymbol{\textit{o}}_{2}\cos(\boldsymbol{\theta}_{1}-\boldsymbol{\delta}_{1})}{\sin\boldsymbol{\theta}_{1}}\right) = -\left(\frac{\boldsymbol{\theta}_{1}}{l_{1}}\right)^{2}\frac{d\boldsymbol{\textit{0}}}{ds}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_{1}-\boldsymbol{\delta}_{1})^{-}} \quad (4.45)$$

Now, the values of $(d\mathbf{0}/ds)$, $(d^2\mathbf{0}/ds^2)$ and $(d^3\mathbf{0}/ds^3)$ will be calculated for spline segments at junction points and these values will be equated with the corresponding values for linear segments for ensuring C^3 continuity at junctions. Using (4.12), the derivative of the spline parameter u with respect to the toolpath length s at the junction point u = 0 can be expressed as

$$\frac{du}{ds}\Big|_{u=0} = \frac{1}{\left(\sqrt{\left(\frac{dP_x}{du}\right)^2 + \left(\frac{dP_y}{du}\right)^2 + \left(\frac{dP_z}{du}\right)^2}\right)}\Big|_{u=0}$$
(4.46)

Differentiating (4.4) with respect to the spline parameter u and substituting u = 0,

$$\frac{dP}{du}\Big|_{u=0} = \begin{bmatrix} -10\\ 10\\ 0\\ 0\\ 0\\ 0\\ 0\\ 0 \end{bmatrix} \begin{bmatrix} P_{1}\\ P_{2}\\ P_{3}\\ P_{4}\\ P_{5}\\ P_{6} \end{bmatrix}^{T} = 10(P_{1} - P_{0})$$
(4.47)

Taking modulus on both the sides,

$$\left(\left\|\frac{d\boldsymbol{P}}{du}\right\|\right)\Big|_{u=0} = \left(\sqrt{\left(\frac{dP_x}{du}\right)^2 + \left(\frac{dP_y}{du}\right)^2 + \left(\frac{dP_z}{du}\right)^2}\right)\Big|_{u=0} = 10(\|\boldsymbol{P_1} - \boldsymbol{P_0}\|) = 10(0.5l) = 5l$$

(4.48)

Similarly,

$$\left(\left\|\frac{d\boldsymbol{P}}{d\boldsymbol{u}}\right\|\right)\Big|_{\boldsymbol{u}=1} = 5l\tag{4.49}$$

Combining (4.46), (4.48) and (4.49), we can write

$$\left. \frac{du}{ds} \right|_{u=0} = \frac{du}{ds} \right|_{u=1} = \frac{1}{5l} \tag{4.50}$$

From (4.11), we have

$$\frac{d^2 u}{ds^2}\Big|_{u=0} = \frac{d^2 u}{ds^2}\Big|_{u=1} = 0; \qquad \frac{d^3 u}{ds^3}\Big|_{u=0} = \frac{d^3 u}{ds^3}\Big|_{u=1} = 0$$
(4.51)

Using (4.50), (4.51) and the method of partial differentiation, the first, second and the third derivative of toolpath orientation vector $\boldsymbol{0}$ with respect to the toolpath length *s* for normalised heptic spline profile at the junction point $\boldsymbol{\theta} = (\theta_1 - \delta_1)^+$ can be written as

$$\frac{d\boldsymbol{0}}{ds}\Big|_{\theta=(\theta_1-\delta_1)^+} = \frac{d\boldsymbol{0}}{ds}\Big|_{u=0} = \left(\frac{d\boldsymbol{0}}{du}\left(\frac{du}{ds}\right)\right)\Big|_{u=0} = \frac{1}{5l}\frac{d\boldsymbol{0}}{du}\Big|_{u=0}$$
(4.52)

$$\frac{d^{2}\boldsymbol{0}}{ds^{2}}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_{1}-\boldsymbol{\delta}_{1})^{+}} = \frac{d^{2}\boldsymbol{0}}{ds^{2}}\Big|_{u=0} = \left(\frac{d^{2}\boldsymbol{0}}{du^{2}}\left(\frac{du}{ds}\right)^{2} + \frac{d\boldsymbol{0}}{du}\left(\frac{d^{2}u}{ds^{2}}\right)\right)\Big|_{u=0} = \left(\frac{d^{2}\boldsymbol{0}}{du^{2}}\left(\frac{du}{ds}\right)^{2}\right)\Big|_{u=0}$$
$$= \frac{1}{25l^{2}}\frac{d^{2}\boldsymbol{0}}{du^{2}}\Big|_{u=0}$$
(4.53)

$$\frac{d^{3}\boldsymbol{O}}{ds^{3}}\Big|_{\theta=(\theta_{1}-\delta_{1})^{+}} = \frac{d^{3}\boldsymbol{O}}{ds^{3}}\Big|_{u=0} = \left(\frac{d^{3}\boldsymbol{O}}{du^{3}}\left(\frac{du}{ds}\right)^{3} + 3\frac{d^{2}\boldsymbol{O}}{du^{2}}\left(\frac{du}{ds}\right)\left(\frac{d^{2}u}{ds^{2}}\right) + \frac{d\boldsymbol{O}}{du}\left(\frac{d^{3}u}{ds^{3}}\right)\right)\Big|_{u=0}$$
$$= \left(\frac{d^{3}\boldsymbol{O}}{du^{3}}\left(\frac{du}{ds}\right)^{3}\right)\Big|_{u=0} = \frac{1}{125l^{3}}\frac{d^{3}\boldsymbol{O}}{du^{3}}\Big|_{u=0}$$
(4.54)

From equations (4.43), (4.44), (4.45), (4.52), (4.53) and (4.54), it can be written that the C^3 continuity will be maintained at the starting junction u = 0 if

$$\frac{d\boldsymbol{0}}{ds}\Big|_{\theta=(\theta_1-\delta_1)^-} = \frac{d\boldsymbol{0}}{ds}\Big|_{\theta=(\theta_1-\delta_1)^+} \quad \Rightarrow \quad \frac{\theta_1}{l_1}\left(\frac{-\boldsymbol{0}_1\cos\delta_1 + \boldsymbol{0}_2\cos(\theta_1-\delta_1)}{\sin\theta_1}\right) = \frac{1}{5l}\frac{d\boldsymbol{0}}{du}\Big|_{u=0}$$
(4.55)

$$\frac{d^{2}\boldsymbol{\theta}}{ds^{2}}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_{1}-\boldsymbol{\delta}_{1})^{-}} = \frac{d^{2}\boldsymbol{\theta}}{ds^{2}}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_{1}-\boldsymbol{\delta}_{1})^{+}} \quad \Rightarrow \quad -\left(\frac{\boldsymbol{\theta}_{1}}{l_{1}}\right)^{2}\left(\frac{\boldsymbol{\theta}_{1}\sin\boldsymbol{\delta}_{1}+\boldsymbol{\theta}_{2}\sin(\boldsymbol{\theta}_{1}-\boldsymbol{\delta}_{1})}{\sin\boldsymbol{\theta}_{1}}\right) = \frac{1}{25l^{2}}\frac{d^{2}\boldsymbol{\theta}}{du^{2}}\Big|_{u=0}$$

$$(4.56)$$

$$\frac{d^3\boldsymbol{\textit{0}}}{ds^3}\Big|_{\boldsymbol{\theta}=(\boldsymbol{\theta}_1-\boldsymbol{\delta}_1)^+} = \frac{d^3\boldsymbol{\textit{0}}}{ds^3}\Big|_{u=0} \quad \Rightarrow \quad -\left(\frac{\boldsymbol{\theta}_1}{l_1}\right)^3 \left(\frac{-\boldsymbol{\textit{o}}_1\cos\boldsymbol{\delta}_1 + \boldsymbol{\textit{o}}_2\cos(\boldsymbol{\theta}_1-\boldsymbol{\delta}_1)}{\sin\boldsymbol{\theta}_1}\right) = \frac{1}{125l^3} \frac{d^3\boldsymbol{\textit{0}}}{du^3}\Big|_{u=0}$$

(4.57)

Now, differentiating the orientation vector $\boldsymbol{0}$ defined by equations (4.32) and (4.33) with respect to the spline parameter u and substituting u = 0, we get

$$\frac{d\boldsymbol{0}}{d\boldsymbol{u}}\Big|_{\boldsymbol{u}=\boldsymbol{0}} = \begin{bmatrix} -14\cos(\delta_1 - \gamma_1) \\ 14 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{0}_0 \\ \boldsymbol{0}_1 \\ \boldsymbol{0}_2 \\ \boldsymbol{0}_3 \\ \boldsymbol{0}_4 \\ \boldsymbol{0}_5 \\ \boldsymbol{0}_6 \\ \boldsymbol{0}_7 \\ \boldsymbol{0}_8 \end{bmatrix}^T, \qquad (4.58)$$

$$\frac{d^{2}\boldsymbol{O}}{du^{2}}\Big|_{u=0} = \begin{bmatrix} -140\cos(\delta_{1}-\gamma_{1})-84\cos(\delta_{1}-\beta_{1})+588(\cos(\delta_{1}-\gamma_{1}))^{2}-196\\140-392\cos(\delta_{1}-\gamma_{1})\\84\\0\\0\\0\\0\\0\\0\end{bmatrix} \begin{bmatrix}\boldsymbol{O}_{0}\\0\\0\\0\\0\\0\\0\end{bmatrix}^{T}\\\boldsymbol{O}_{0}\\0\\0\\0\\0\end{bmatrix} \begin{bmatrix}\boldsymbol{O}_{0}\\0\\0\\0\\0\\0\\0\end{bmatrix} \end{bmatrix}^{T} and$$

$$(4.59)$$

$$\frac{d^{3}\boldsymbol{0}}{du^{3}}\Big|_{u=0} = \begin{bmatrix} 588(-20\cos(\delta_{1}-\gamma_{1})-6\cos(\delta_{1}-\beta_{1})+42(\cos(\delta_{1}-\gamma_{1}))^{2}-11) \\ 1848-3528\cos(\delta_{1}-\gamma_{1}) \\ 420 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{0}_{0} \\ \boldsymbol{0}_{2} \\ \boldsymbol{0}_{3} \\ \boldsymbol{0}_{4} \\ \boldsymbol{0}_{5} \\ \boldsymbol{0}_{6} \\ \boldsymbol{0}_{7} \\ \boldsymbol{0}_{8} \end{bmatrix}^{T} \\ (4.60)$$

Where the expression for μ_1 is given by

$$\mu_{1} = -84(-273\cos(\delta_{1} - \gamma_{1}) + 22\cos(\delta_{1} - \beta_{1}) + 5\cos(\delta_{1} - \alpha_{1}) + 42\cos(\gamma_{1} - \beta_{1}) - 126\cos(\delta_{1} - \gamma_{1})\cos(\delta_{1} - \beta_{1}) - 210(\cos(\delta_{1} - \gamma_{1}))^{2} + 490(\cos(\delta_{1} - \gamma_{1}))^{3} + 70)$$
(4.61)

Thus, from equations (4.55), (4.56) and (4.57) it can be concluded that the C^3 continuity will be maintained at the junction corresponding to u = 0 if

$$\begin{cases} \frac{\theta_1}{l_1} \left(\frac{-\boldsymbol{o_1} \cos \delta_1 + \boldsymbol{o_2} \cos(\theta_1 - \delta_1)}{\sin \theta_1} \right) = \frac{1}{5l} \frac{d\boldsymbol{0}}{du} \Big|_{u=0} \\ - \left(\frac{\theta_1}{l_1} \right)^2 \left(\frac{\boldsymbol{o_1} \sin \delta_1 + \boldsymbol{o_2} \sin(\theta_1 - \delta_1)}{\sin \theta_1} \right) = \frac{1}{25l^2} \frac{d^2 \boldsymbol{0}}{du^2} \Big|_{u=0} \\ - \left(\frac{\theta_1}{l_1} \right)^3 \left(\frac{-\boldsymbol{o_1} \cos \delta_1 + \boldsymbol{o_2} \cos(\theta_1 - \delta_1)}{\sin \theta_1} \right) = \frac{1}{125l^3} \frac{d^3 \boldsymbol{0}}{du^3} \Big|_{u=0} \end{cases}$$
(4.62)

Where the expressions for $(d\mathbf{0}/du)$, $(d^2\mathbf{0}/du^2)$ and $(d^3\mathbf{0}/du^3)$ at the junction u = 0 are presented in (4.58), (4.59) and (4.60) respectively. Similarly, the conditions for C^3 continuity at the junction corresponding to u = 1 are formulated as

$$\begin{cases} \frac{\theta_2}{l_2} \left(\frac{\boldsymbol{o}_3 \cos \delta_2 - \boldsymbol{o}_2 \cos(\theta_2 - \delta_2)}{\sin \theta_2} \right) = \frac{1}{5l} \frac{d\boldsymbol{O}}{du} \Big|_{u=1} \\ - \left(\frac{\theta_2}{l_2} \right)^2 \left(\frac{\boldsymbol{o}_3 \sin \delta_2 + \boldsymbol{o}_2 \sin(\theta_2 - \delta_2)}{\sin \theta_2} \right) = \frac{1}{25l^2} \frac{d^2 \boldsymbol{O}}{du^2} \Big|_{u=1} \\ - \left(\frac{\theta_2}{l_2} \right)^3 \left(\frac{\boldsymbol{o}_3 \cos \delta_2 - \boldsymbol{o}_2 \cos(\theta_2 - \delta_2)}{\sin \theta_2} \right) = \frac{1}{125l^3} \frac{d^3 \boldsymbol{O}}{du^3} \Big|_{u=1} \end{cases}$$
(4.63)

where the expressions for $(d\boldsymbol{0}/du)$, $(d^2\boldsymbol{0}/du^2)$ and $(d^3\boldsymbol{0}/du^3)$ at the junction u = 1 are formulated as

$$\frac{d\boldsymbol{0}}{d\boldsymbol{u}}\Big|_{\boldsymbol{u}=1} = \begin{bmatrix} & 0 \\ & 0$$

(4.64)

Where the expression for μ_2 is given by

$$\mu_{2} = 84(-273\cos(\delta_{2} - \gamma_{2}) + 22\cos(\delta_{2} - \beta_{2}) + 5\cos(\delta_{2} - \alpha_{2}) + 42\cos(\gamma_{2} - \beta_{2})$$

- 126\cos(\delta_{2} - \gamma_{2})\cos(\delta_{2} - \beta_{2}) - 210(\cos(\delta_{2} - \gamma_{2}))^{2} + 490(\cos(\delta_{2} - \gamma_{2}))^{3}
+ 70)

It should be noted here that two more conditions are required to evaluate the exact locations of the remaining 8 orientation control points. This will be done by formulating user-defined orientation error tolerance limit constraint and position and orientation spline synchronization constraint as discussed in the following sections.

4.3.2.2 Pre-defined orientation error tolerance limit, ε_{ori}

The normalized B-spline curve defined by equation (4.32) and (4.33) is so formulated that the maximum orientation error e_{max} occurs at the mid-point of the spline. Substituting u = 0.5 in equation (4.32), we get

The maximum orientation error e_{max} must satisfy the user-defined orientation error tolerance limit ε_{ori} and is evaluated as

$$e_{max} = \cos^{-1}(\boldsymbol{0}(0.5), \boldsymbol{0}_4) \le \varepsilon_{ori} \tag{4.68}$$

Taking the limiting case,

$$\cos(\varepsilon_{orl}) = \mathbf{0}(0.5) \cdot \mathbf{0}_{4} = \frac{\mathbf{B}(0.5) \cdot \mathbf{0}_{4}}{\|\mathbf{B}(0.5)\|}$$

$$= \frac{\left(\frac{1}{64}\mathbf{0}_{1} + \frac{3}{32}\mathbf{0}_{2} + \frac{15}{64}\mathbf{0}_{3} + \frac{5}{16}\mathbf{0}_{4} + \frac{15}{64}\mathbf{0}_{5} + \frac{3}{32}\mathbf{0}_{6} + \frac{1}{64}\mathbf{0}_{7}\right) \cdot \mathbf{0}_{4}}{\|\mathbf{B}(0.5)\|}$$

$$= \frac{\mathbf{0}_{1} \cdot \mathbf{0}_{4} + 6(\mathbf{0}_{2} \cdot \mathbf{0}_{4}) + 15(\mathbf{0}_{3} \cdot \mathbf{0}_{4}) + 20(\mathbf{0}_{4} \cdot \mathbf{0}_{4}) + 15(\mathbf{0}_{5} \cdot \mathbf{0}_{4}) + 6(\mathbf{0}_{6} \cdot \mathbf{0}_{4}) + \mathbf{0}_{7} \cdot \mathbf{0}_{4}}{64 \|\mathbf{B}(0.5)\|}$$

$$= \frac{\cos \gamma_{1} + 6\cos \beta_{1} + 15\cos \alpha_{1} + 20 + 15\cos \alpha_{2} + 6\cos \beta_{2} + \cos \gamma_{2}}{64 \|\mathbf{B}(0.5)\|}$$

$$(4.69)$$

where the angles $\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1$ and γ_2 are defined in section 4.3.2 and shown in figure 4.9.

4.3.2.3 Synchronisation of position and orientation spline

The synchronization of the position and orientation spline at the start and end of the spline curve can be achieved if following relations are respected (figure 4.10).

$$Start: \frac{\delta_1}{\theta_1} = \frac{2.5l}{l_1} \tag{4.70}$$

$$End: \ \frac{\delta_2}{\theta_2} = \frac{2.5l}{l_2} \tag{4.71}$$

However, as we need only one additional condition to solve for orientation control points, equations (4.70) and (4.71) are combined as

$$\frac{\delta_1}{\theta_1} l_1 = \frac{\delta_2}{\theta_2} l_2 \tag{4.72}$$



Figure 4.10 Synchronization of position and orientation spline

Due to the non-linear nature of constraint equations, closed form solution for angles $\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1$ and δ_2 does not exist. Thus, Newton-Raphson iterative optimization algorithm is utilized. To implement the algorithm, the 8 constraint equations formulated in (4.62), (4.63), (4.69) and (4.72) are first expressed in the form:

$$f(\mathbf{x}) = f(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2) = 0 \Rightarrow \begin{cases} f_1(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2) = 0\\ f_2(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2) = 0\\ \dots\\ f_8(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2) = 0 \end{cases}$$
(4.73)

The derivative of this system is 8×8 jacobian matrix given by

$$\boldsymbol{J}(\boldsymbol{x}) = \boldsymbol{J}(\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2) = \begin{bmatrix} \frac{\partial f_1}{\partial \alpha_1} & \cdots & \frac{\partial f_1}{\partial \delta_2} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_8}{\partial \alpha_1} & \cdots & \frac{\partial f_8}{\partial \delta_1} \end{bmatrix}$$

The primary limitation of the Newton-Raphson algorithm comes from improper choice of initial estimate. It is observed that the algorithm converges fast if the initial estimates are chosen as $\alpha_{1} = \alpha_{2} = \varepsilon$

$$\beta_{1} = \beta_{2} = 2\varepsilon_{ori}$$

$$\beta_{1} = \beta_{2} = 2\varepsilon_{ori}$$

$$\gamma_{1} = \gamma_{2} = 3\varepsilon_{ori}$$

$$\delta_{1} = \delta_{2} = 3.5\varepsilon_{ori}$$

$$\therefore \mathbf{x}^{(0)} = [\varepsilon_{ori}, \varepsilon_{ori}, 2\varepsilon_{ori}, 2\varepsilon_{ori}, 3\varepsilon_{ori}, 3\varepsilon_{ori}, 3.5\varepsilon_{ori}]^{T}$$

$$(4.75)$$

The subsequent iteration is computed by maximizing the second order Taylor approximation of f(x) about $x^{(k)}$ and it follows following iteration rule.

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} - J(\mathbf{x}^{(k-1)})^{-1} f(\mathbf{x}^{(k-1)}) \quad , \quad k \ge 1$$
(4.76)

The convergence criteria given by equation (4.77), is a reasonable choice for computational complexity and accuracy of the solution.

$$\|\Delta \mathbf{x}\| = \left\| \mathbf{x}^{(k)} - \mathbf{x}^{(k-1)} \right\| \le 10^{-8} \tag{4.77}$$

It should be noted here that the synchronization constraints corresponding to equations (4.70) and (4.71) are yet to be satisfied separately. In section (4.2), the length of segment P_1P_2 is calculated as

$$l_{old} = min\left\{\frac{4 \varepsilon_{pos}}{3 \cos\left(\frac{\alpha}{2}\right)}, \frac{l_1}{5}, \frac{l_2}{5}\right\}$$
(4.78)

77

(4.74)

Using (4.70) and (4.71), the new length l_{new} of segment P_1P_2 can be calculated as

$$l_{new} = \frac{\delta_1}{2.5\theta_1} l_1 = \frac{\delta_2}{2.5\theta_2} l_2 \tag{4.79}$$

If l_{new} is smaller than l_{old} , the locations of position control points need to be recalculated based on l_{new} using equations (4.21) and (4.23). Otherwise, if l_{old} is smaller than l_{new} , the locations of all the orientation control points should be updated. For this, the values of δ_1 and δ_2 are reevaluated based on equations (4.70) and (4.71) using l_{old} . Keeping δ_1 constant, α_1 , β_1 and γ_1 are calculated using equation (4.62). The value of angle α_2 , β_2 and γ_2 are calculated using eq. (4.63).

4.4 Feedrate planning

After geometrical modification of the toolpath, feedrate planning along the toolpath is carried out. The smooth toolpath can be sub-grouped into two types of segments, i.e. (1) modified linear segments and (2) inserted spline segments. For feedrate planning, the lengths of toolpath segments need to be calculated. As discussed in section 4.2, the linear segments p_1P_0 and p_3P_6 are calculated as $(l_1 - 2.5l)$ and $(l_2 - 2.5l)$ respectively (figure 4.11).



Figure 4.11 Lengths of toolpath segments

The length of the spline segment is calculated using Simpson's rule with an adaptive bisection technique as proposed by Yuen et.al. [16]. The length of spline segment from u = a to u = b is given by

$$l(b) - l(a) = \int_{a}^{b} \frac{ds}{du} du$$
(4.80)

From equations (4.12) and (4.80),

$$l(b) - l(a) = \int_{a}^{b} \left(\sqrt{\left(\frac{dP_x}{du}\right)^2 + \left(\frac{dP_y}{du}\right)^2 + \left(\frac{dP_z}{du}\right)^2} \right) du = \int_{a}^{b} f(u) \, du \tag{4.81}$$

From Simpson's rule and equation (4.81),

$$l(b) - l(a) = \int_{a}^{b} f(u) \, du = \frac{b - a}{6} \left[f(a) + 4f\left(\frac{a + b}{2}\right) + f(b) \right]$$
(4.82)

The interval [a, b] is divided into two equal segments $[a_1, b_1]$ and $[a_2, b_2]$, and the lengths of the two new intervals are approximated with equation (4.82). Given some tolerance ε_l , the approximation of the length over [a, b] is within tolerance if equation (4.83) is satisfied. If equation (4.83) is violated, the interval [a, b] is divided into two equal segments and the approximation procedure is repeated until all segments respect (4.83).

$$|l(a_1, b_1) + l(a_2, b_2) - l(a, b)| < \varepsilon_l$$
(4.83)

After calculating toolpath segments lengths, a C^3 continuous cubic acceleration feedrate profile with constant interpolation time t_s is used to generate displacement command s(t) as a function of time t as shown in figure 4.12. Cubic acceleration feedrate profile comprised of 7 sections. During time intervals T_1 , T_2 and T_3 the tool accelerates from starting feedrate, f_s to the commanded feedrate f_c . This is followed by a constant velocity region where the feedrate is kept constant at f_c for time intervals T_4 . Lastly, the tangential feedrate decreases to reach the end feedrate, f_e within the time intervals T_5 , T_6 and T_7 . The maximum allowable acceleration, deceleration and jerk along the toolpath are represented as A_{max} , D_{max} and J_{max} respectively. The mathematical formulations 79



of the analytical expressions governing the values of time intervals can be found in [44].

Figure 4.12 Cubic acceleration feedrate profile

For linear toolpath segments, the tool tip position command, p(t) = [x(t), y(t), z(t)] and orientation command, $o(t) = [o_i(t), o_j(t), o_k(t)]$ share linear relationship with the displacement command s(t) for interpolation between the cutter location (CL) data $[p_1, o_1]$ and $[p_2, o_2]$ as shown in equation (4.84) and (4.85).

$$\boldsymbol{p}(t) = \boldsymbol{p}_1 + \frac{(\boldsymbol{p}_2 - \boldsymbol{p}_1)}{l_1} \boldsymbol{s}(t); \tag{4.84}$$

$$\boldsymbol{\theta}(t) = \frac{\boldsymbol{\theta}_1 \sin(\theta_1 - \theta(t)) + \boldsymbol{\theta}_2 \sin\theta(t)}{\sin\theta_1}, \quad \theta(t) = \frac{\theta_1}{l_1} s(t)$$
(4.85)

Where $l_1 = ||\mathbf{p}_2 - \mathbf{p}_1||$ and $\theta_1 = \cos^{-1}(\mathbf{o}_1, \mathbf{o}_2)$. However, interpolating position and orientation command along a spline toolpath segment is more complex as the spline parameter u does not share any direct relationship with the displacement command s(t). Various works are presented in literature for interpolating position and orientation commands for spline toolpath segment as shown in figure 4.13.

(i) Natural Interpolation

$$s(t) \qquad \qquad \underbrace{\frac{l_k}{S_k}} \quad u(t) \qquad \qquad \underbrace{s(t)} \quad F = A_k u^5 + B_k u^4 + \dots + F_k \qquad \qquad \underbrace{x(t)} \quad y(t)$$

(ii) Interpolation Using Taylor Series Expansion

$$s(t) \qquad u_i = u_{i-1} + \dot{u}_i T_s + \ddot{u}_i T_s^2 \qquad u(t) \qquad \text{Spline} \qquad x(t) \\ \dot{u} = u_s \dot{s}, \ddot{u} = u_{ss} \dot{s}^2 + u_s \ddot{s} \qquad \text{Toolpath} \qquad y(t) \qquad \text{Spline} \qquad y(t) \qquad$$

(iii) Interpolation with Feedrate Correction Polynomial

s(t)	Feed Correction Polynomial:	$\hat{u}(t)$	Spline	x(t)
	$u = A_k^f s^7 + B_k^f s^6 + \dots + H_k^f$		Toolpath	<i>y</i> (<i>t</i>)

Figure 4.13 Different methods of interpolating along a given spline toolpath

The first approach, also called as natural interpolation, is based on optimizing the toolpath geometry to yield minimum discrepancy between the spline parameters and the arc length increments, resulting in an Optimally Arc Length Parameterized (OAP) spline [44]. However, if

there is significant discrepancy between the spline parameter and arc length, this discrepancy is directly reflected as fluctuation in the feedrate profile.



Figure 4.14 Adaptive approach of formulating feed correction spline u(s)

In literature, this feedrate fluctuation is mitigated by applying Taylor series interpolation according to which the new value of the spline parameter is calculated by expanding its time-dependent Taylor series around the value of spline parameter in the previous interpolation time. Applying Taylor series interpolation results in significant improvement in feedrate consistency. However, it is computationally complex. The third approach is based on scheduling of the spline parameter to yield desired arc displacement, hence the desired feedrate profile accurately. This is done by either approximating the relationship between the arc length and the spline parameter using a feedcorrection polynomial or by solving the exact value of the spline parameter using an iterative interpolation technique. However, sometimes it is difficult to express the spline parameter u as a function of arc length s using the same feed-correction polynomial throughout the spline segment within specified prediction error tolerance limits. To overcome this problem, Yuen et.al. [18] suggested an adaptive approach to formulate a 9th order feed-correction spline to approximate the non-linear relation between spline parameter and displacement along the tool path. For our purpose, it is sufficient to use the feed-correction spline formulation technique used by Yuen et al [18] to express spline parameter u as a function of toolpath length s. The flowchart describing the formulation of feed correction spline u(s) is presented in figure 4.14. Inverse kinematics is then performed on the interpolated position and orientation data $[x(t), y(t), z(t), O_i(t), O_i(t), O_k(t)]^T$ to generate axis motion commands $[X(t), Y(t), Z(t), \theta_p(t), \theta_s(t)]^T$. The velocity, acceleration and the jerk profiles are generated using axis motion commands for all the five axes. In case drive velocity, acceleration or jerk limits are violated, commanded feedrate f_c is adjusted using equation 4.86 as discussed by Xavier et. al. [15].

$$f_{c}' = \min_{i=1..5} \left(f_{c}, \frac{V_{max}^{i}}{|q_{s}^{i}|}, \sqrt{\frac{A_{max}^{i}}{|q_{ss}^{i}|}}, \sqrt[3]{\frac{J_{max}^{i}}{|q_{sss}^{i}|}} \right)$$
(4.86)

Where q_{s}^{i} , q_{sss}^{i} , q_{sss}^{i} are axis geometrical derivatives with respect to displacement *s* along the toolpath and V_{max}^{i} , A_{max}^{i} , J_{max}^{i} are axes velocity acceleration and jerk limits. The position and orientation data are interpolated again using the same procedure with the modified commanded feedrate f_{c}^{\prime} .

4.5 Experimental results

The proposed corner smoothing algorithm which leads to improvements in cycle time, tracking accuracy and smoothness while respecting user-defined error tolerance limits, is validated by conducting experiments on a Fadal 2216 vertical machining center retrofitted with a Nikken 5AX-130 compact tilting rotary table. The kinematic configuration of the 5-axis CNC system is shown in figure 4.15.



Figure 4.15 Kinematic configuration of 5-axis CNC machining center

The inverse kinematic kinematics transformation from the work-piece coordinate system to the machine co-ordinate system is given by

$$\begin{cases} X = -\cos(C) P_{x} - \sin(C) P_{y} \\ Y = \cos(A) \sin(C) P_{x} - \cos(A) \cos(C) P_{y} - \sin(A) P_{z} - \sin(A) L_{ac,z} \\ Z = \sin(A) \sin(C) P_{x} - \sin(A) \cos(C) P_{y} + \cos(A) P_{z} + \cos(A) P_{z} + \cos(A) L_{ac,z} + L_{Tya,z} \\ A = \cos^{-1} O_{k} \\ C = \cos^{-1} (O_{i}/O_{j}) \end{cases}$$

Where $L_{ac,z}$ = 70 mm and $L_{Tya,z}$ = 150 mm are the offsets, determined by the geometry of the rotary table. The machine is controlled by in-house developed, open real-time CNC, allowing for the rapid implementation of controllers and trajectories. The experimental test bed is shown in the figure 4.16.



Figure 4.16 Experimental test bed consisting of the Fadal VMC 2216, Nikken 130-AX compact tilting table and dSPACE open real-time controller

In order to design controller for the five- axis system, the drive inertia, viscous friction and static friction parameters are identified using rigid body identification procedure as described by Erkorkmaz et.al. [46]. The values of the identified parameters corresponding to each axis are presented in table 4.1.

Parameters	X-Axis	Y-Axis	Z-Axis	A-Axis	C-Axis
Amplifier gain	6.5723	6.2274	6.4841	1.2700	1.2700
$K_a(A/V)$					
Motor torque constant	0.4769	0.4769	0.4769	0.3333	0.3333
$K_t(Nm/A)$					
Lead screw gain	1.5915	1.5915	1.2732	0.0056	0.0111
r_g	(mm/rad)	(mm/rad)	(mm/rad)	(rad/rad)	(rad/rad)
Equivalent inertia	7.0028e-3	8.1904e-3	7.6715e-3	8.1068e-5	2.2122e-4
$J(kg.m^2)$					
Viscous damping	0.023569	0.043009	0.032328	0.001100	0.002900
$B(kg.m^2/s)$					
Positive coul. friction	1.83	1.17	2.3	0.2147	0.1568
$T_{coul}^+(Nm)$					
Negative coul. friction	-1.1048	-1.02	-2.7	-0.2707	-0.1430
$T_{coul}^{-}(Nm)$					

 Table 4.1 Plant parameters for Fadal VMC 2216

The drives are controlled by implementing standard proportional, integral and derivative (PID) controllers at a sampling frequency of 1 kHz. The PID gains are configured so that all the axes have matching dynamics in order to minimize the contour errors. The gain values of the designed PID control law are presented in table 4.2. A schematic diagram showing the control law and modelled plant with identified parameters are shown in figure 4.17.

Parameters	X-Axis	Y-Axis	Z-Axis	 Parameters	A-Axis	C-Axis
$K_p(V/mm)$	10	12.344	13.880	 $K_p(rad/mm)$	245.55	335.03
$K_i(V/mm/s)$	50	61.718	69.400	 $K_i(rad/mm/s)$	1227.7	1675.1
$K_d(V.s/mm)$	0.3	0.3670	0.4147	 $K_d(rad.s/mm)$	7.0147	9.5925



Figure 4.17 Control law and modelled plant for five-axis corner smoothing experiments

The blade toolpath for five-axis corner smoothing experiments is shown in figure 4.18. A close observation of the experimental toolpath reveals that the G01 data points are moderately spaced in the beginning of the path which is followed by zone 1 where a large gap in the toolpath can be observed. This is followed by a region of high curvature and closely spaced G01 data points.



Figure 4.18 Test toolpath for five-axis corner smoothing experiments



Figure 4.19 Toolpath position and orientation error in reference command generation

As all the position and orientation corners in the toolpath are smoothed by inserting position and orientation micro-splines respectively, the proposed smoothing algorithm leads to position and orientation error at and around each CAM transition points as shown in figure 4.19. From the figure, it is observed that the tool tip position and orientation error shows various peaks. These peak points correspond to the mid-point of the inserted spline segment at each CAM transition point as the splines are formulated to have maximum error at the mid-points (see figure 4.3). The symmetrical nature of the error plot at each transition point can be attributed to the symmetrical nature of the inserted spline. Moreover, it can be observed that pattern of both position and orientation error plot are similar. This validates proper synchronization of the position and orientation splines as discussed in section 4.3.2.3. The position and orientation error always remain within the pre-defined corner position and orientation tolerance of 0.1 mm and 0.1 degree respectively. However, it should be noted that at most of the transition points, the error remain well below the pre-defined tolerance limits. This is because the CAM points are very closely spaced resulting in very small toolpath segment lengths. As each linear toolpath segment is used to smooth two corner points corresponding to the start and end-point of the segment, the maximum length of the segment used in each smoothing must be below half of the total toolpath segment length as also shown in equation 4.28.

To show the effect of the proposed algorithm in cycle time reduction, the experiments are carried out before and after smoothing the toolpath. A commanded feedrate of $100 \text{ }mm/sec}$ is used to carry out the experiments. Cubic acceleration feedrate profile is used in both cases. Corner position and orientation tolerance limits are chosen as 0.1 mm and pi/1800 rad respectively. Axis feedrate, acceleration and jerk limits are set as 200 mm/sec, $1000 \text{ }mm/sec^2$ and $50000 \text{ }mm/sec^3$ respectively for all the translational drives and 1 rad/sec, $10 \text{ }rad/sec^2$ and $500 \text{ }rad/sec^3$ respectively for all the rotational drives to ensure that the manufacturer specified drive limits are not violated. The tangential feedrate profile for proposed corner smoothing algorithm is also compared against the predicted tangential feedrate profile of Mori Seiki NMV5000 CNC system for the same toolpath and commanded feedrate.



Figure 4.20 Experimental tangential feedrate profile before and after corner smoothing

The tangential feedrate profiles calculated using the position data from encoder measurements are shown in figure 4.20. From the tangential feedrate profile, it can be observed the tool cannot reach the commanded feedrate of 100 mm/sec in any of the three cases. This is because the CAM data points are very closely spaced and toolpath has high curvature which results in lower tooltip velocity in order to ensure the feedrate, acceleration and jerk limits of the drive. The tool tip velocity increases at zone 1 as the toolpath segment length is comparatively large. As the tool leaves zone 1, it enters a region of high curvature closely spaced CAM points which results in decrease in the tangential feedrate as shown. It is also observed that the cycle time before implementation of smoothing algorithm in exact-stop mode is 8.288 sec. However, after implementing the algorithm cycle time gets reduced to 2.849 sec which leads to a total cycle time reduction of about 66% in this case. The observed cycle time reduction is due to two reasons: (1) The proposed algorithm avoids the tool motion to cease at each CAM transition point. (2) The maximum feedrate achieved after implementation of the smoothing algorithm is higher than the exact stop case which results in further improvement in cycle time. When compared against the predicted tangential feedrate profile of Mori Seiki NMV5000 CNC system, it is observed that the feedrate profile with proposed corner smoothing algorithm leads to slight improvement in cycle time. From this, it can be concluded that the proposed five-axis corner smoothing algorithm results in reduction of cycle time.



Figure 4.21 Jerk of axis reference commands for experiments

Next, the smoothness of the proposed C^3 continuous corner smoothing algorithm is validated. The smoothness of the commanded toolpath motion can be observed by analyzing the axis reference jerk commands which can be evaluated by utilizing the method of digital differentiation as shown in equation 4.87.

$$\frac{d^{3}\boldsymbol{q}}{dt^{3}} \approx \frac{-\boldsymbol{q}(t_{i+3}) + 8\boldsymbol{q}(t_{i+2}) - 13\boldsymbol{q}(t_{i+1}) + 13\boldsymbol{q}(t_{i-1}) - 8\boldsymbol{q}(t_{i-2}) + \boldsymbol{q}(t_{i-3})}{8\Delta t^{3}} \ \forall (t_{i+1} - t_{i}) = \Delta t$$

$$(4.87)$$

Where $\boldsymbol{q} = [X Y Z A C]^T$ and $\Delta t = 0.001 \, sec$. The reference jerk commands are calculated for both C^2 and C^3 continuous cases for all the five drives and plotted as shown in figure 4.21. From the plot, it can be observed that the reference jerk corresponding to zone 1 is comparatively lower for all the drives in both C^2 and C^3 continuous cases as predicted. As already mentioned zone 1 is followed by a region of high curvature and closely spaced CAM data points demanding higher jerk from all the drives. After further analysis, it is observed that the jerk profiles for the proposed C^3 continuous trajectory remains continuous throughout the toolpath motion. The C^2 continuous trajectory however, results in discontinuities with higher amplitudes in jerk profile for all the five axes especially after zone 1 where the tool enters region of high curvature and closely spaced CAM points. This discontinuities and higher jerk amplitudes might result in tool vibration and poor surface quality.



Figure 4.22 Experimental average axis tracking error from encoder measurements

Axis	X-Axis	Y-Axis	Z-Axis	C-Axis	A-Axis
C^3 continuous	41.78 μm	29.25 µm	8.50 µm	0.28 mrad	0.21 mrad
C^2 continuous	49.89 μm	36.53 µm	10.41 μm	0.31 mrad	0.26 mrad
% improvement	16.26%	19.93%	18.34%	8.09%	18.58%

Table 4.3 Percentage improvement in tracking error

In addition to showing improvements in cycle time and toolpath smoothness, it is observed that the C^3 continuous trajectory shows improvement in average axis tracking error for all the drives when compared against the C^2 continuous trajectory as shown in figure 4.22. Table 4.3 shows the average tracking error for all the axes in C^2 and C^3 continuous cases. Among the translational axes, the reference jerk commands for X-drive is the most aggressive followed by Y-drive. It should also be noted that the reference jerk commands for C-drive is more aggressive than A-drive. This might be a possible reason for higher axis tracking error shown by X-axis and C-axis as compared to other translational and rotational axes respectively.

4.6 Summary

In this chapter, a novel decoupled approach is introduced to achieve local corner smoothing of five-axis toolpath. Toolpath position and orientation are smoothed by inserting quintic and normalized septic micro-splines, respectively between the adjacent linear toolpath segments. Optimal control points are calculated for position and orientation splines to maintain C^3 continuity throughout the toolpath while respecting user-defined corner position tolerance (ε_{pos}) and orientation tolerance (ε_{ori}) limits. Synchronization of position and orientation splines is carried out. After geometrical modification of the toolpath, feedrate planning is performed using cubic acceleration feedrate profile to preserve C^3 continuity in toolpath motion. The linear and the spline sections of the toolpath are interpolated by linear and spline interpolation techniques respectively. The non-linear relation between spline parameter(u) and displacement along the toolpath(s) is approximated using ninth-order feed correction spline. Inverse kinematics is then performed on the interpolated position and orientation data to generate axis motion commands. In case drive

acceleration or jerk limits are violated, commanded feedrate is adjusted and position and orientation data are interpolated again. The proposed C^3 continuous toolpath motion is compared against the unsmooth and C^2 continuous motion in experiments to show improvements in cycle time, tracking accuracy and smoothness throughout the toolpath.

Chapter 5: Conclusions and future work

5.1 Conclusions

In this thesis, a new cycle time prediction scheme is proposed for machining operations based on identified CNC machine dynamics in exact-stop and continuous modes. Various system identification techniques are utilized to identify the implemented trajectory generation and corner smoothing technique and feed drive dynamics of the CNC system. It is shown that the cycle time of NC machining process is predominantly effected by trajectory generation and corner smoothing techniques implemented on the CNC system. However, the closed-loop feed drive dynamics does not have much influence on the cycle time. An analytical approach for predicting cycle time based on the identified CNC system dynamics and given part program is presented. In addition, a novel decoupled approach for five-axis corner smoothing is presented to reduce the cycle time of milling operations. Toolpath position and orientation are smoothed by inserting quintic and normalized septic micro-splines, respectively between the adjacent linear toolpath segments. Optimal control points are calculated for position and orientation splines to achieve C^3 continuity at the junctions between the splines and the linear segments while respecting user-defined corner position tolerance and orientation tolerance limits. Synchronization of position and orientation splines is carried out. After geometrical modification of the toolpath, feedrate planning is performed using C^3 continuous cubic acceleration feedrate profile to ensure a smooth and continuous motion throughout the five-axis toolpath.

The primary results of the thesis work are summarized as follows:

(1) The proposed cycle time prediction algorithm is found to estimate feedrate profile and cycle time for a given toolpath accurately when compared against the experimentally measured data from Mori Seiki NMV5000 DCG CNC Machining Center. The cycle time prediction error remains within 5% for various 2-axis, 3-axis and 5-axis toolpaths in both exact- stop and continuous mode. (2) The proposed C^3 continuous five-axis corner smoothing algorithm is employed to generate the smoothed toolpath motion which is compared against the unsmooth and C^2 continuous motion in simulations and experiments. Up to 67% reduction in cycle time and consistent improvement in smoothness and axis-tracking error are achieved using the proposed algorithm for five-axis machining of free-form surfaces.

The contributions made by this thesis work are summarized below:

- (1) An attempt is made to identify the trajectory generation and corner smoothing algorithm applied in a commercial CNC machine with the help of simple G-code command and various existing system identification techniques. A new technique is developed to predict cycle time of milling operations involving 2-axis, 3-axis and 5-axis toolpaths robustly. None of the other existing cycle time prediction technique can be applied to all range of milling operations to predict cycle time within acceptable prediction error.
- (2) The three-axis position corner smoothing technique available in literature ensures C^2 continuity of the toolpath. However, high quality standards requirement for manufacturing products leads to the growing demand for smoother toolpath generation with continuous jerk. A new local toolpath position smoothing technique is developed to ensure jerk continuous motion with minimal feed fluctuation and user-defined position tolerance.
- (3) A novel technique of tool orientation smoothing by inserting normalized heptic B-spline between linearly interpolated orientation vectors is developed. Normalized spline has never been used before in the literature to the best of the author's knowledge. Optimal control points for orientation spline is calculated to ensure the modified path satisfies user-defined tool orientation error and jerk continuity at the connection between the linearly interpolated orientation vector and orientation spline.

5.2 Future research directions

Several research directions can be pursued in future in the field of cycle time prediction and fiveaxis corner smoothing. Some of them are discussed as follows:

- (1) It is observed that the cycle time prediction error increases when the commanded toolpath is very small and the commanded feedrate is very high especially in continuous mode. This might be because in the current research, a simplified model of the system is assumed. A more detailed study of the CNC system can be carried out with the developed algorithm as a base for the purpose of improving the results in above-specified case. Also, as most of the machining operations in industry also involves auxiliary motions such as tool-change time, approach time etc. In the current research all the auxiliary motions are ignored as they generally account for less than 5% of total cycle time. In order to implement the developed approach in shop floor environment, a framework needs to be developed to include both cutting and non-cutting time.
- (2) The developed five-axis corner smoothing approach ensures 3rd order continuity in toolpath geometry and motion and it leads to significant reduction in cycle time and tracking error. However, the cycle time can be reduced further by optimizing the curvature of the toolpath. Minimum curvature five-axis toolpath can be developed which can ensure fastest possible cycle time. The problem is certainly mathematically challenging but it has huge potential for practical application. The other potential improvement can be brought upon by developing a five-axis corner smoothing algorithm which can be implemented on real-time. CNC research in the field of toolpath smoothing is yet to address this issue.
- R.T. Coelho, A.F.D. Souza, A.R. Roger, A.M.Y. Rigatti, A.A.D.L. Ribeiro, "Mechanistic approach to predict real machining tme for miling free-form geometries applying high feed rate," *International Journal of Advanced Manufacturing Technology*, vol. 46, pp. 1103-1111, 2010.
- [2] X. Yan, K. Shirase, M. Hirao and T. Yasui, "NC program evaluator for higher machining productivity," *International Journal of Machine tool and Manufacture*, vol. 39, pp. 1563-1573, 1999.
- [3] X. Yan, K. Shirase and H. Kawabata, "Improving productivity of machining canters based on NC program diagnostic system," *Journal of Japan Society for Precision Engineering*, vol. 63, no. 7, pp. 1044-1048, 1997.
- [4] R. Othmani, W. Bouzid and M. Hbaieb, "Machining time in rough milling," *Materials Technology*, vol. 23, pp. 169-173, 2008.
- [5] M. Hbaieb, R. Othmani and W. Bouzid, "Time modeling in high-speed machining of mold pockets," *International Journal of Advanced Manufacturing Technology*, vol. 53, pp. 113-120, 2011.
- [6] J. Y. Jung, "Manufacturing cost estimation for machined parts based on manufacturing features," *Journal of Intelligent Manufacturing*, vol. 13, pp. 227-238, 2002.
- [7] A. Dugas, J.J. Lee, M. Terrier and J.Y. Hascoet, "Development of a machining simulator considering machine behaviour," *Journal of Engineering Manufacture*, vol. 217, pp. 1333-1339, 2003.

- [8] X. Yan, K. Shirase, M. Hirao and T. Yasui, "NC program evaluator for higher machining productivity," *International Journal of Machine Tools and Manufacture*, vol. 39, pp. 1563-1573, 1999.
- [9] W. Mou, W. Chu and L. Tang, "Reseach on accurate simulation of NC program machining time," *Tool Technology*, vol. 43, pp. 50-54, 2009.
- [10] B.S. So, Y.H. Jung, J.W. Park and D.W. Lee, "Five axis machining time estimation algorithm based on machine characteristics," *Journal of Materials Processing Technology*, vol. 187, pp. 37-40, 2007.
- [11] M. Monreal and C.A. Rodriguez, "Influence of toolpath strategy on the cycle time of high speed milling," *Computer-Aided Design*, vol. 35, pp. 395-401, 2003.
- [12] H. Siller, C.A. Rodriguez and H. Ahuett, "Cycle time prediction in high speed milling operations for sculptured surface finishing," *Journal of Materials Processing Technology*, vol. 174, pp. 355-362, 2006.
- [13] E.Y. Heo, D.W. Kim, B.H. Kim and F.F. Chen, "Estimation of NC machining time using NC block distribution for sculptured surface machining," *Robotics and Computer-Integrated Manufacturing*, vol. 22, pp. 437-446, 2006.
- [14] M.C. Ho, Y.R. Hwang and C.H. Hu, "Five-axis orientation smoothing using quaternion interpolation algorithm," *International Journal of Machine Tools and Manufacture*, vol. 43, pp. 1259-1267, 2003.
- [15] X. Beudaert, P.Y. Pechard and C. Tournier, "5-Axis toolpath smoothing based on drive constraints," *International journal of machine tools and manufacture*, pp. 958-965, 2011.

- [16] W. Lei, M. Sung, L. Lin and J. Huang, "Fast real-time NURBS path interpolation for CNC machine tools," *International journal of machine tools and manufacture*, vol. 47, no. 10, pp. 1530-1541, 2007.
- [17] R.V. Fleisig and A.D. Spence, "A constant feed and reduced angular acceleration interpolation algorithm for multi-axis machining," *Computer-Aided Design*, vol. 33, pp. 1-15, 2001.
- [18] A. Yuen, K. Zhang and Y. Altintas, "Smooth trajectory generation for five-axis machine tools," *International journal of machine tools and manufacture*, pp. 11-19, 2013.
- [19] V. Pateloup, E. Duc and P. Ray, "Bspline approximation of circle arc and straight line for pocket machining," *Computer-Aided Design*, vol. 42, pp. 817-827, 2010.
- [20] H. Zhao, L.M. Zhu and H. Ding, "A real-time look-ahead interpolation methodology with curvature-continuous B-spline transition scheme for CNC machining of short line segments," *International Journal of Machine Tools & Manufacture*, vol. 65, pp. 88-98, 2013.
- [21] Q. Bi, Y. Wang, L. Zhu and H. Ding, "A practical continuous-curvature Bezier transition algorithm for high speed machining of linear toolpath," *Intelligent Robotics and Applications*, vol. 7102, pp. 465-476, 2011.
- [22] S.J. Yutkowitz, and W. Chester, "Apparatus and method for smooth cornering in a motion control system". United States Patent 6922606, 2005.
- [23] K. Erkorkmaz, C.H. Yeung and Y. Altintas, "Virtual CNC system. Part II. High speed contouring application," *International Journal of Machine Tools & Manufacture*, vol. 46, pp. 1124-1138, 2006.

- [24] B. Sencer and E. Shamoto, "Curvature-continuous sharp corner smoothing scheme for cartesian motion systems".
- [25] C.A. Ernesto and R.T. Farouki, "High-speed cornering by CNC machines under prescribed bounds on axis accelerations and toolpath contour error," *International Journal of Advanced Manufacturing Technology*, vol. 58, pp. 327-338, 2012.
- [26] B. Sencer and E. Shamoto, "Curvature continuous sharp corner smoothing scheme for cartesian motion systems," 2014.
- [27] X. Beudaert, S. Lavernhe and C. Tournier, "5-axis local corner rounding of linear toolpath discontinuities," *International journal of machine tools and manufacture*, pp. 9-16, 2013.
- [28] B. Hurst, The journeyman guides to CNC machines, Lulu Press, 2006.
- [29] Mori Seiki Co.Ltd., "DMG Mori," [Online]. Available: http://www.moriseiki.it.
- [30] Mori Seiki Co.Ltd., "NMV500 DCG," [Online]. Available: http://www.moriseiki.co.uk/2007/05/nmv5000-dcg%E2%84%A2-the-new-benchmark-inflexibility-and-perfection/.
- [31] D. Mori, "MAPPS III DMG Mori," [Online]. Available: http://www.moriseiki.co.uk/products/applications/mapps-iii/.
- [32] Fanuc Corporation, Fanuc 31i Parameter Manual, Fanuc Corporation.
- [33] N. Borlin, "Non-linear Optimization: Trust-region methods," December 2007. [Online]. Available: https://www8.cs.umu.se/kurser/5DA001/HT07/lectures/trust-handouts.pdf.
- [34] G. A. Shultz, R. B. Schnabel and R.H. Byrd, "A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties," *SIAM Journal of Numerical Analysis*, vol. 22, pp. 47-67, 1985.

- [35] R.H. Byrd, R.B. Schnabel, and G.A. Shultz, "Approximate Solution of the Trust Region Problem by Minimization over Two-Dimensional Subspaces," *Mathematical Programming*, vol. 40, pp. 247-263, 1988.
- [36] Wikipedia, "Linear lesat squares (Mathematics)," 2014. [Online]. Available: http://en.wikipedia.org/wiki/Linear_least_squares_(mathematics).
- [37] B. Sencer and Y. Altintas, "Identification of 5-axis machine tools feed drive systems for contouring simulation," *International Jornal of Automation Technology*, vol. 5, pp. 377-386, 2011.
- [38] K. Erkorkmaz and W. Wong, "Rapid identification techniques for for virtual CNC drives," *International Journal of MAchine Tools and Manufacture*, vol. 47, pp. 1381-1392, 2007.
- [39] DMG Mori Seiki Co. Ltd., NMV5000 DCG User Manual.
- [40] Rik Pintelon and Johan Schoukens, System Identification: A Frequency Domain Approach (2nd Ed), vol. 76, John Wiley and Sons, Inc., 2012.
- [41] R. Aarts, "System identification and parameter estimation," 2012. [Online]. Available: http://www.utwente.nl/ctw/wa/web_dev/old/lectures/113170/notes/notes.pdf.
- [42] B. Sencer, Y. Altintas and E. Croft, "Feed optimization for five-axis CNC machine tools with drive constraints," *Machine Tools and Manufacture*, pp. 733-745, 2008.
- [43] E.B. Dam, M. Koch and M. Lillholm, "Quaternions, Interpolation and Animation," 12 July 1998. [Online]. Available: http://web.mit.edu/2.998/www/QuaternionReport1.pdf.
- [44] "Rodrigues' Rotation Formula," [Online]. Available: http://en.wikipedia.org/wiki/Rodrigues'_rotation_formula.

- [45] K. Erkorkmaz, "Optimal trajectory generation and precision tracking control for multi-axis machine," The university of British columbia, 2004.
- [46] K. Erkorkmaz and Y. Altintas, "Quintic spline interpolation with minimal feed fluctuation," *Journal of manufacturing science and engineering*, pp. 339-349, 2005.
- [47] K. Erkorkmaz and Y. Altintas, "High speed CNC system design. Part II: Modeling and identification of feed drives," *International Journal of Machine Tools and Manufacture*, vol. 41, pp. 1487-1509, 2001.
- [48] A. Yuen, "Spline interpolation and contour error pre-compensation for five-axis machining," The University of British Columbia, 2011.