

---


Electronic Theses and Dissertations, 2004-2019

---

2014

## Modeling and Contour Control of Multi-Axis Linear Driven Machine Tools

Ran Zhao  
*University of Central Florida*

 Part of the [Mechanical Engineering Commons](#)  
Find similar works at: <https://stars.library.ucf.edu/etd>  
University of Central Florida Libraries <http://library.ucf.edu>

This Doctoral Dissertation (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations, 2004-2019 by an authorized administrator of STARS. For more information, please contact [STARS@ucf.edu](mailto:STARS@ucf.edu).

---

### STARS Citation

Zhao, Ran, "Modeling and Contour Control of Multi-Axis Linear Driven Machine Tools" (2014). *Electronic Theses and Dissertations, 2004-2019*. 4834.  
<https://stars.library.ucf.edu/etd/4834>

MODELING AND CONTOUR CONTROL OF MULTI-AXIS LINEAR DRIVEN  
MACHINE TOOLS

By

RAN ZHAO

B.S. Hefei University of Technology, 2004

M.S. University of Science and Technology of China, 2008

M.S. University of Central Florida, 2013

A dissertation submitted in partial fulfillment of the requirements  
for the degree of Doctor of Philosophy  
in the Department of Mechanical & Aerospace Engineering  
in the College of Engineering & Computer Science  
at the University of Central Florida  
Orlando, Florida

Spring Term  
2014

Major Professor: Chengying Xu

© 2014 Ran Zhao

## ABSTRACT

In modern manufacturing industries, many applications require precision motion control of multi-agent systems, like multi-joint robot arms and multi-axis machine tools. Cutter (end effector) should stay as close as possible to the reference trajectory to ensure the quality of the final products. In conventional computer numerical control (CNC), the control unit of each axis is independently designed to achieve the best individual tracking performance. However, this becomes less effective when dealing with multi-axis contour following tasks because of the lack of coordination among axes. This dissertation studies the control of multi-axis machine tools with focus on reducing the contour error. The proposed research explicitly addresses the minimization of contour error and treats the multi-axis machine tool as a multi-input-multi-output (MIMO) system instead of several decoupled single-input-single-output (SISO) systems. New control schemes are developed to achieve superior contour following performance even in the presence of disturbances. This study also extends the applications of the proposed control system from plane contours to regular contours in  $\mathbb{R}^3$ . The effectiveness of the developed control systems is experimentally verified on a micro milling machine.

## ACKNOWLEDGMENTS

First and foremost, I would like to give my sincere appreciations to my advisor Prof. Chengying Xu for her encouragements and supports during the last five years of my study. I have been truly motivated by her dedication and passion for academic researches. None of these researches would be possible without her insights and guidance.

I would also like to thank Prof. Lin, Prof. An, Prof. Das and Prof. Bai for serving as my advisory committee members and for providing all the instructions, comments and help on my research. Their rich experiences, profound academic background and valuable insights become the lighthouse guiding me through this research.

I want to give my special thanks to my wife for her unconditional love and support. Finally, I dedicate this dissertation to my parents, who always encourage me and never lose faith in me. Your love is the biggest fortune in my life.

# TABLE OF CONTENTS

LIST OF FIGURES .....	viii
LIST OF TABLES .....	xi
CHAPTER 1 INTRODUCTION.....	1
1.1 Reference Trajectory Optimization.....	2
1.1.1 Jerk Limited Feedrate Profile and Spline Interpolation .....	2
1.1.2 Quintic Spline Interpolation .....	4
1.2 Individual Axis Control.....	5
1.3 Contour Error and Contour Error Estimation .....	8
1.4 Two Contour Control Configurations .....	11
1.4.1 Feedforward Type Contour Control.....	11
1.4.2 Feedback Type Contour Control .....	13
1.5 Dissertation Outline .....	13
CHAPTER 2 MODELING OF MACHINE TOOLS FOR LOW SPEED MOTIONS.....	15
2.1 Experimental Setup.....	15
2.2 Dynamics of Experimental Setup .....	19
2.3 Identification of Force Ripple.....	21
2.4 Identification of Rigid Body Dynamics .....	28
2.5 Identification of Frictions Using Luge Model.....	32

2.5.1	Cost Functions .....	33
2.5.2	Particle Swarm Optimization (PSO).....	34
2.5.3	Parameter Identification .....	35
2.6	Experimental Validations .....	40
2.6.1	Closed Loop Test .....	40
2.6.2	Tracking Test.....	42
2.7	Summary .....	44
CHAPTER 3 ROBUST CONTROL FOR PLANAR CONTOUR FOLLOWING .....		45
3.1	Discrete Sliding Mode Control (DSMC) .....	45
3.2	Discrete Sliding Mode Contour Control (DSMCC) for Nominal System .....	47
3.3	DSMCC for Robust Performance .....	54
3.4	Experimental Validations .....	55
3.4.1	Performance Index .....	56
3.4.2	Experiment 1 Guitar Contour .....	56
3.4.3	Experiment 2 Dog Bone Contour .....	59
3.4.4	Experiment 3 Performance with Perturbations and Uncertainties.....	61
3.5	Summary .....	64
CHAPTER 4 MOVING FRAME OPTIMIZATION OF REGULAR CURVES IN $\mathbb{R}^3$ .....		66
4.1	Background Knowledge .....	66

4.2	Problem Formulation .....	68
4.2.1	Local Frame of Regular Contours in $\mathbb{R}^3$ .....	68
4.2.2	Optimization of Local Frame .....	70
4.3	Contour Error Estimation of $\mathbb{R}^3$ Contours in Local Frame .....	74
4.4	Controller Design.....	78
4.5	Robust Contour Control .....	79
4.6	Experimental Validations .....	83
4.6.1	Contour Design.....	83
4.6.2	Experiment 1 Contour Following for Curves of Large Torsion.....	84
4.6.3	Experiment 2 Performance of Full Controller .....	88
4.7	Summary .....	89
CHAPTER 5 FUTURE WORK .....		90
LIST OF REFERENCES .....		92



## LIST OF FIGURES

Figure 1: Comparison between trapezoidal and jerk limited feedrate profile .....	3
Figure 2: Quintic spline interpolation .....	4
Figure 3: Decouple control for a bi-axial contour following .....	7
Figure 4: Comparison between estimated contour error and actual contour error .....	8
Figure 5: Feedforwr type contour control (CCC) .....	12
Figure 6: Block diagram of feedback type contour control.....	13
Figure 7: 3-axis micro mill .....	16
Figure 8: Block diagram of experimental machine tools .....	16
Figure 9: Quadrant glitch caused by frictions .....	18
Figure 10: Block diagram of dynamics of direct drive .....	20
Figure 11: Block diagram of modified dynamics of direct drive .....	20
Figure 12: Experiment setup for force ripple test .....	22
Figure 13: Control signal showing force ripple #1 .....	22
Figure 14: Control signal showing force ripple #2 .....	23
Figure 15: Control signal showing force ripple #3 .....	23
Figure 16: Spectrum of force ripple 3 .....	24
Figure 17: Identification results of $g_2$ and $g_6$ .....	26
Figure 18: Comparison between actual force ripple and simulated force ripple .....	26
Figure 19: Feedforward control .....	27
Figure 20: Control signal of PI controller without feedforward compensation .....	27
Figure 21: (a) Signal of PI controller (b) Signal of feedforward compensator .....	28

Figure 22: Excitation signal.....	29
Figure 23: (a) Measured and simulated velocities (b) Error.....	32
Figure 24: Convergence of cost function of static parameters .....	36
Figure 25: Static speed-friction mapping and results of identification.....	37
Figure 26: Excitation signal for dynamic parameter identification .....	38
Figure 27: Convergence of cost function of dynamic parameters .....	39
Figure 28: Experimental and simulated responses.....	40
Figure 29: Closed loop step response.....	42
Figure 30: Tracking error comparison .....	43
Figure 31: Control signal comparison.....	44
Figure 32: Cutaway guitar contour .....	57
Figure 33: Contour error of guitar contour at $20mm/s$ .....	57
Figure 34: Contour error of guitar contour at $30mm/s$ .....	58
Figure 35: Dog bone contour.....	59
Figure 36: Contour error of dog bone contour at $20mm/s$ .....	60
Figure 37: Contour error of Dog bone contour at $30mm/s$ .....	61
Figure 38: Circular contour following under nominal condition.....	62
Figure 39: Circular contour following with a $4Kg$ payload.....	63
Figure 40: Circular contour following with large step disturbance .....	64
Figure 41: Local frame of three dimensional contours .....	69
Figure 42: Schematic diagram of the local frame rotation .....	73
Figure 43: Schematic diagram of equivalent contour error in $\mathbb{R}^3$ .....	76

Figure 44: Block diagram of the robust control.....	80
Figure 45: Reference trajectory for each feed drive .....	83
Figure 46: Optimizing rotation angle $\theta$ .....	85
Figure 47: Cost functions of optimized and non-optimized move frames .....	85
Figure 48: Control signal using non-optimized moving frame .....	87
Figure 49: Control signal using optimized moving frame .....	88
Figure 50: Performance comparison between the optimized and non-optimized frames .....	88
Figure 51: Contour error using robust inner loop and improved contour error model .....	89

## LIST OF TABLES

Table 1: Identification results of force ripples.....	25
Table 2: Identification results of rigid body dynamics .....	31
Table 3: Identification results of static parameters .....	37
Table 4: Identification results of dynamics parameters .....	39
Table 5: Guitar contour results .....	58
Table 6: Dog bone contour results .....	61
Table 7: Controller gains of contour control in $\mathbb{R}^3$ .....	84

## CHAPTER 1 INTRODUCTION

More than two decades ago, the decline of manufacturing industries of the United State had concerned many researchers. They argued that the manufacturing productivities are critical to the country's competitiveness and living standard. After the recent economic downturn, manufacturing industry has become a key part to revive the nation's economy and to bring jobs back to the U.S. In the author's opinion, machine tools play a critical role to improve the productivity and quality of manufacturing industries. The importance of machine tools lay in the fact that many fundamental manufacturing processes would require state-of-the-art machine tools. It is simply impossible to have world-class manufacturing without world-class tools.

It has been reported that the positioning error of machine tool feed drives can contribute up to 90% of the geometry error of the final products [1] and therefore the demands for motion control system with ever higher performance never ceased. The significance of motion control goes beyond just the machining tools. In fact, many newly developed technologies, for example, atomic force microscope (AFM), fabrication of polymer-derived ceramics (PDC) [2] and  $\mu$ -Rd deposition [3], are based off precision motion systems.

In modern industries, the machining of complex geometries and sculptured surfaces requires simultaneous control of multiple axes. In such applications, two major sources of performance degradations are 1) perturbations including modeling uncertainties as well as external disturbances 2) and lack of coordination among axes. And these problems became the motive of this research. In the following two sections, we first introduce several motion control technologies and researches previously developed.

## 1.1 Reference Trajectory Optimization

### 1.1.1 Jerk Limited Feedrate Profile and Spline Interpolation

Due to the computational limit, CNCs were only capable of constant velocity feedrate [4]. With the developments of micro-controller and digital signal processor (DSP), jerk limited interpolations have been developed for smoother acceleration and de-acceleration process [5] than constant velocity feedrate profile. A comparison between trapezoidal feedrate profile and jerk limited feedrate profile is illustrated in Figure 1. Compared to the trapezoidal feedrate, the jerk limited feedrate profile contains less high frequency components than the trapezoidal feedrate profile does, which is beneficial to the tracking accuracy.

Another important task is *interpolation*, which refers to the generation of the points constituting the reference trajectory [6]. Conventional CNC systems are mostly based on basic linear and circular interpolations [7, 8]. Linear interpolation connects two points using a linear toolpath while in the circular interpolation, the objective becomes moving the tool along a circular path connecting the control points.

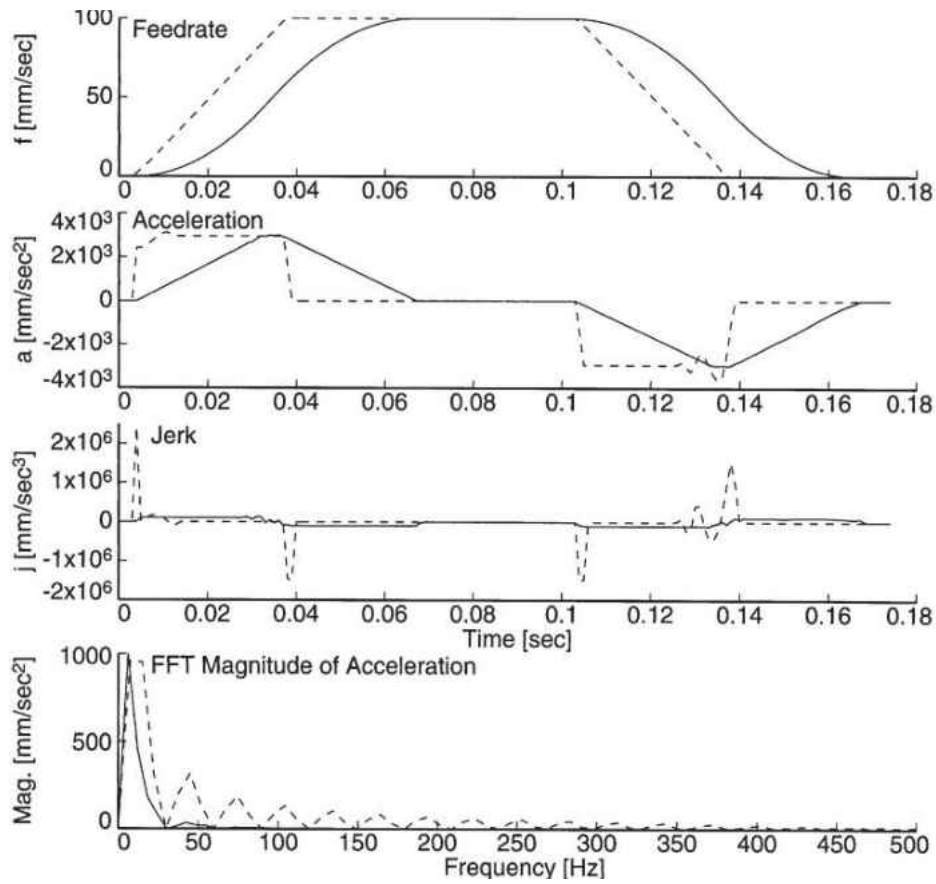


Figure 1: Comparison between trapezoidal and jerk limited feedrate profile [9]

The advances in aerospace and medical industries boost the needs for products of complex geometries. These products, for examples, man-made joints, require the machine to cut sculpture-like surfaces. Though a complex toolpath can be approximated by a series of linear and circular segments, the optimal results cannot be achieved. The bandwidth of machine tools is limited by factors like power of actuators, noise level of the encoder signal and delay in the feedback loop. Linear and circular interpolations cause discontinuities causing actuator saturation and excite unmolded structural dynamics. An optimized reference trajectory should be smooth and mainly consists of low frequency components [9]. Extensive studies have been reported on the trajectory optimizations, including using B-Spline [10] and quintic spline [7] to

generate smooth contours. Here we briefly introduce the quintic spline interpolation because part of the results will be used in the later part of the dissertation. Interested readers are referred to [6, 9] for more information. For simplicity, we only introduce the results on plane quintic splines. The results can be easily extended to  $\mathbb{R}^3$ .

### 1.1.2 Quintic Spline Interpolation

The quintic spline interpolation connects  $N+1$  knots using  $N$  fifth order splines, see Figure 2. The resulted spline is to ensure that continuity up to the second order derivative is preserved along the overall composite curve.

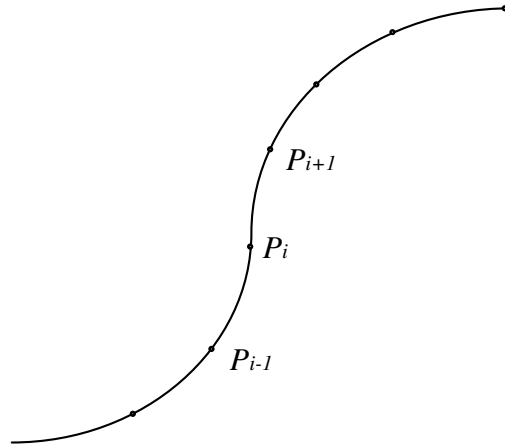


Figure 2: Quintic spline interpolation

The quintic spline is parameterized using the chord length between two consecutive knots. The fitting procedure starts with calculating the first and second derivatives for the  $i$ th knot  $P_i$  by fitting a cubic polynomial

$$\begin{cases} x(u) = a_{xk}u^3 + b_{xk}u^2 + c_{xk}u + d_{xk} \\ y(u) = a_{yk}u^3 + b_{yk}u^2 + c_{yk}u + d_{yk} \end{cases} \quad (1)$$



through  $P_{i-2}, P_{i-1}, P_i, P_{i+1}$ . The parameter of the cubic polynomial can be uniquely determined using the coordinates of four consecutive points  $P_{i-2}, P_{i-1}, P_i, P_{i+1}$ . Once the cubic fitting is obtained the first derivative (tangent vector)  $\mathbf{t}_i = [t_{ix} \ t_{iy}]^T$  and second derivative (normal vector)  $\mathbf{n}_i = [n_{ix} \ n_{iy}]^T$  for  $P_i$  can be computed. With the first and second derivatives of  $P_i$ , a quintic spline can be fit between two consecutive knots  $P_i, P_{i+1}$

$$\begin{cases} x(u) = A_{xk}u^5 + B_{xk}u^4 + C_{xk}u^3 + D_{xk}u^2 + E_{xk}u + F_{xk} \\ y(u) = A_{yk}u^5 + B_{yk}u^4 + C_{yk}u^3 + D_{yk}u^2 + E_{yk}u + F_{yk} \end{cases} \quad (2)$$

where  $A_{xk}, \dots, F_{xk}, A_{yk}, \dots, F_{yk}$  are spline coefficients and  $u$  is the spline parameters. The boundary conditions used to solve the spline parameters are

$$\begin{cases} x'(0) = t_{ix} & x''(0) = n_{ix} \\ y'(0) = t_{iy} & y''(0) = n_{iy} \\ x'(l_{i,i+1}) = t_{i+1x} & x''(l_{i,i+1}) = n_{i+1x} \\ y'(l_{i,i+1}) = t_{i+1y} & y''(l_{i,i+1}) = n_{i+1y} \\ x(0) = p_{ix} & x(l_{i,i+1}) = p_{i+1x} \\ y(0) = p_{iy} & y(l_{i,i+1}) = p_{i+1y} \end{cases} \quad (3)$$

where  $l_{i,i+1}$  is the chord length between  $P_i, P_{i+1}$ .

## 1.2 Individual Axis Control

The performance of multi-axis machine tools has been mostly improved by enhancing the tracking performance of individual axis. Related topics have been extensively studied over the past decades. For example, if the model of closed loop system is available, zero phase error

tracking algorithm (ZPETC) [11] had been proposed to increase the overall bandwidth and minimize the phase error. When the closed loop system is of minimum phase, the ZPETC utilized the inversion of the system model to achieve a unit gain transfer function. If the system has non-stable zeros, ZPETC is able to cancel the phase error. In addition, repetitive control (RC) [12, 13] and iterative learning control (ILC) [14, 15] also find applications when the reference trajectories or the disturbances are periodic. If the disturbance is not periodic, disturbance observer (DOB) had been proposed to enhance disturbance rejections [16]. In addition to the motion controller design, precision modeling of machine tool feed drives have drawn considerable attentions as well [1, 17-24]. In industry, the loop transmission technology is almost dominant. For other techniques of single axis tracking, interested readers are referred to [25, 26]. It should be noted that these researches are designed for single servomechanism and are normally referred to as decoupled controls because they are essentially local performance optimization and become less effective when multiple axes need to be controlled simultaneously [27]. This point can be further illustrated by Figure 3.

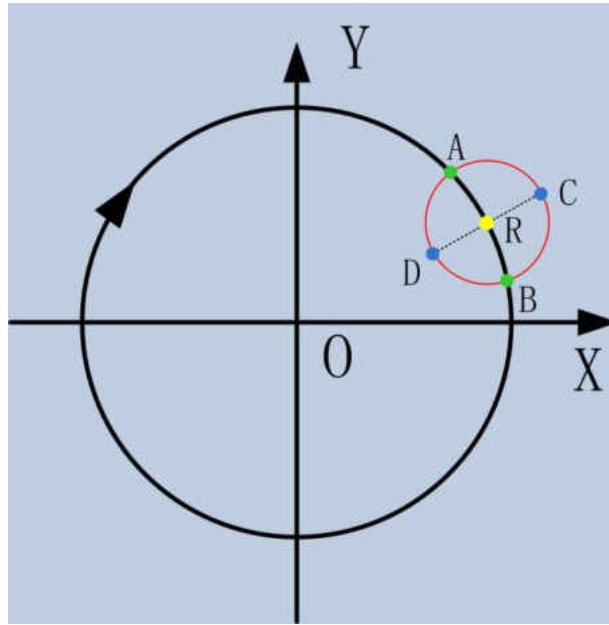


Figure 3: Decouple control for a bi-axial contour following

Assume a two-axis machine tool is commanded to follow a circular toolpath of radius  $\rho_R$  (big circle). A small red circle of radius  $\rho_r$  ( $\rho_r \ll \rho_R$ ) is placed using the current reference point R as its center. A, B, C and D are the four possible locations of the cutter. From the perspective of the decoupled controller, all four points have the same performances since the magnitudes of the tracking error are the same ( $\rho_r$ ). However, one can observe that A and B are far better than C and D as they are still on the desired curve while C and D deviate considerably. It is clear that, due to the lack of coordination, decoupled control is not capable of recognizing the performance of the overall system and as a result, the contour following performance cannot be guaranteed.

### 1.3 Contour Error and Contour Error Estimation

Tracking error and contour error may be the most important two performance indices of machine tools, See Figure 4. Denote the actual position of the cutter as  $Q$  and current commanded position as  $D$ , the tracking error is as

$$E = D - Q \quad (4)$$

In the decoupled design, each controller strives to reduce the magnitude of the tracking error vector  $E$ . However, contour error is more important than the individual tracking error on the accuracy of the multi-axis machining.

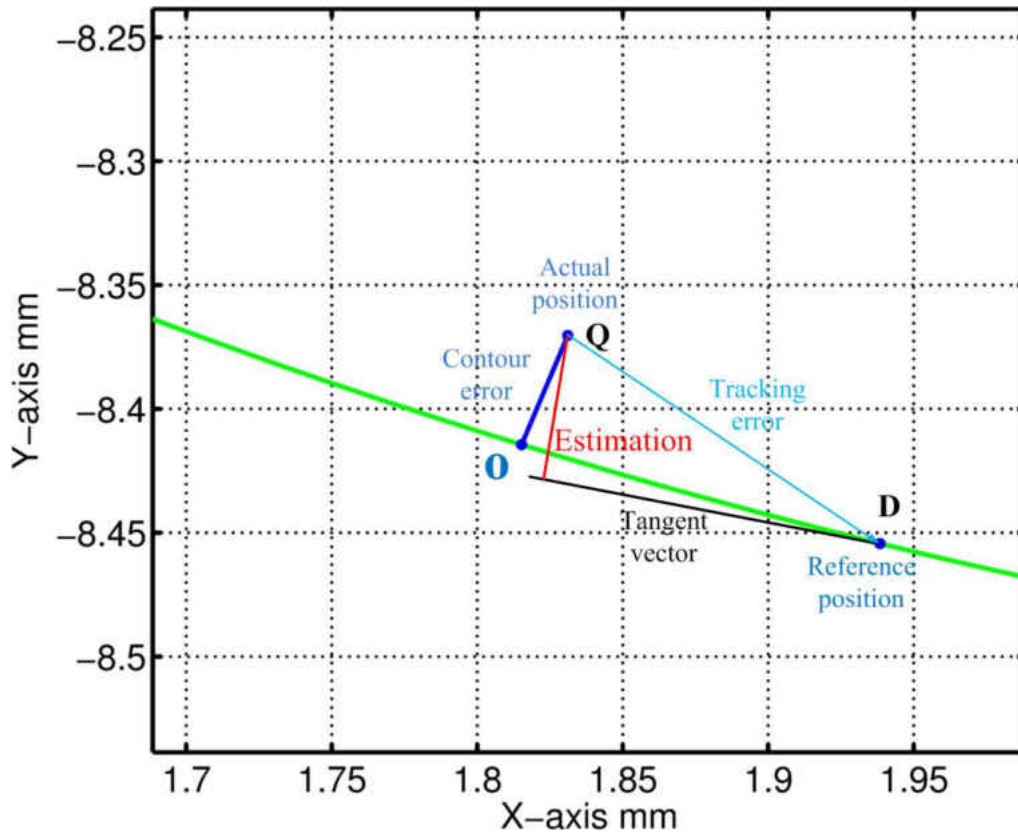


Figure 4: Comparison between estimated contour error and actual contour error

The general definition of *contour error* is the shortest distance between the actual position Q and the desired curve R (solid dark blue line in Figure 4). For linear or circular contours, the contour error can be efficiently computed the using the linear cross coupling gains [28, 29].

$$\varepsilon = -\sin \chi E^X + \cos \chi E^Y \quad (5)$$

The parameter angle  $\chi$  is

$$\sin \chi = -\frac{V^Y}{V_R}, \cos \chi = \frac{V^X}{V_R} \quad (6)$$

where  $V^X$ ,  $V^Y$  and  $V_R$  are desired velocity for X-axis, Y-axis and the synthesized velocity of the desired contour, respectively.  $E^X$  and  $E^Y$  are the tracking error of the X-axis and Y-axis, respectively. For circular contours, the contour error is [28]

$$\varepsilon = -E^X \left( \sin \chi - \frac{E^X}{2R_a} \right) + E^Y \left( \cos \chi + \frac{E^Y}{2R_a} \right) \quad (7)$$

where  $R_a$  is the radius of the circular contour. Eqn (7) is actually a time-varying version of Eqn (5). It should be noted that Eqn (5) and (7) are exact contour error. However, the contour error in general lacks analytical expressions for complex curves and therefore can only be estimated. One approach is to approximate the contours with a series of linear and circular segments and use Eqn (5) and (7) to match the instantaneous segment [30]. Though works have been done on the optimization on the choice of the cross coupling gains [30, 31], the design and analysis of CCC is always a challenging and time-consuming task [32].

Here we give an example of how to calculate the actual contour error of complex curves. The reference curve is given by a plane quintic spline with the following parameters, the definition of these parameters can be found in Eqn (2).

$$\begin{bmatrix} A_{xk} \\ B_{xk} \\ C_{xk} \\ D_{xk} \\ E_{xk} \\ F_{xk} \end{bmatrix} = \begin{bmatrix} -0.0040146 \\ -0.0027863 \\ 0.00740748 \\ 0.05596339 \\ 0.87860363 \\ 1.32000000 \end{bmatrix}, \begin{bmatrix} A_{yk} \\ B_{yk} \\ C_{yk} \\ D_{yk} \\ E_{yk} \\ F_{yk} \end{bmatrix} = \begin{bmatrix} 0.031059 \\ -0.10998 \\ 0.152731 \\ 0.146529 \\ -0.57768 \\ -8.16000 \end{bmatrix}, u \in [0.0268, 1.0370] \quad (8)$$

Assume that the current reference position is  $D = [1.938 \ -8.454]^T$  and the cutter location is  $Q = [1.831 \ -8.370]^T$ , see Figure 4. The actual contour error is determined by finding the minimum of the following function

$$D_{is}(u) = \sqrt{(x(u) - 1.831)^2 + (y(u) + 8.370)^2} \quad (9)$$

on the interval  $u \in [0.0268, 1.0370]$ . The problem can be solved by finding the minimum of a tenth order polynomial on a fixed interval. The actual contour error is determined using the Matlab optimization toolbox and the result is

$$\varepsilon = 0.0468 \quad (10)$$

One can see that the computational loads associated with computing the actual contour error are too high for servo control system when the curves are complex. The sampling rate of modern system is getting ever higher and the algorithm's efficiency must be taken into

consideration. (The sampling rate of modern servo system can be as high as 250 Kilohertz [33-35]).

To circumvent this problem, a variety of contour error estimation methods have been studied [29, 36, 37]. Here we briefly introduce the tangent line estimation method and compare the estimated result to the actual contour error. Still use Figure 4, the estimated contour error is the distance between Q and the tangent line at point D. The estimated contour error, denoted by  $e$ , is

$$e = 0.0517 \quad (11)$$

By comparing the results in Eqns (10), (11), the tangent line estimation clearly introduces an estimation error. However, tangent line approximation has been widely accepted in favor of its computational efficiency.

#### 1.4 Two Contour Control Configurations

The basic idea of contour control is to introduce an extra control effort to reduce the contour error. Depending on how this concept is implemented, contour control can be in general categorized into feedforward type and feedback type.

##### 1.4.1 Feedforward Type Contour Control

The feedforward type contour control has received extensive attentions after Koren proposing his famous cross coupling control (CCC) [8, 25, 28, 38]. The structure of basic CCC is illustrated in Figure 5. A feedback control stabilizing the open loop system is required in the

CCC design. As such, CCC is a feedforward type control and the system is of a dual-controller structure.

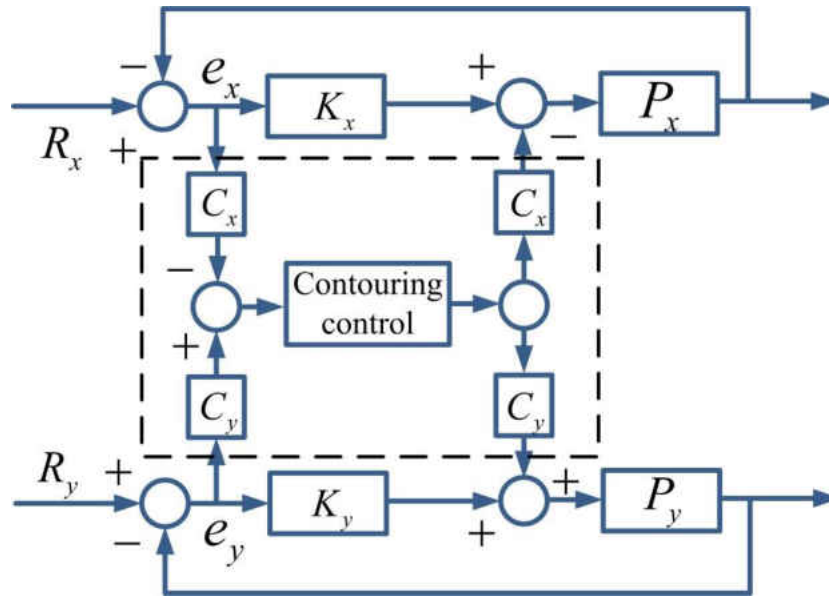


Figure 5: Feedforwrd type contour control (CCC)

The design of the CCC is based on contour error transfer function (CETF) [37, 39, 40]

$$c_e = g_{CETF} c_{e0}$$

where  $c_{e0}$  and  $c_e$  is the contour error without and with CCC, respectively. CETF is designed to have a small low frequency gain so the contour error can be reduced when CCC is implemented. One problem associated with the dual-controller structure is that the two controllers potentially work against each other [41, 42]. CCC may reduce the contour error at a cost of increasing the tracking error and the tracking controller may resist it. As such, it is difficult to distinguish which controller dominates the overall dynamics. Besides, due to the dual controller structure, stability analysis of the CCC is often very complex, especially when multi-axis system is involved or



variable controller gains are needed [27]. As such, the potentials of feedforward type are limited. For example, Lo proposed a CCC for three dimensional contour tracking [43]. Yet the stability of the propose approach was not vigorously.

#### 1.4.2 Feedback Type Contour Control

Compare to the feedforward type contour control, the feedback contour control is typically of much simpler structure. The idea of the feedback type control is to transform the contour error into system state variables and convert the contour following task into regulation problems [44]. Examples of feedback type contour control include synchronization control [45] and task frame based contour control [46-48], see Figure 6.

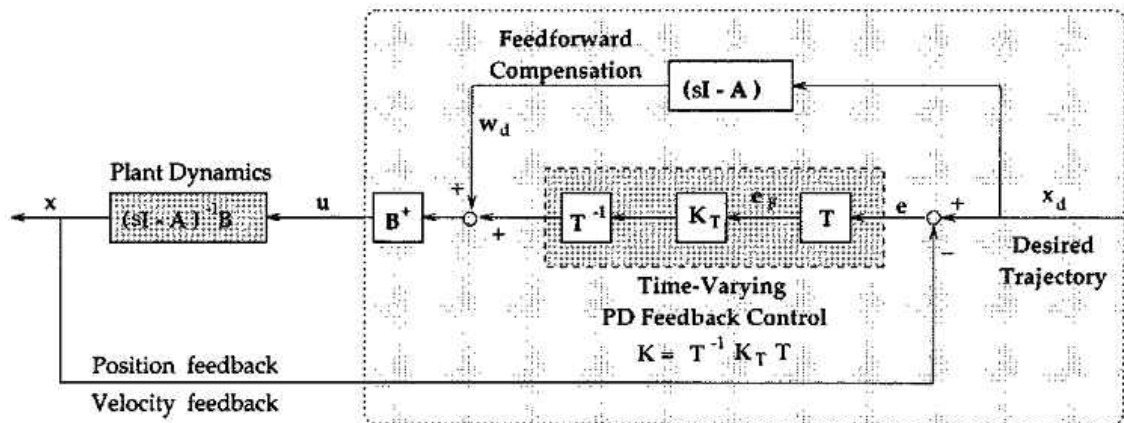


Figure 6: Block diagram of feedback type contour control [47]

#### 1.5 Dissertation Outline

In the second chapter, the experimental machine tool setup is introduced. Modeling works have been designed to obtain a precision model of the machine tool feed drives with focus

on low speed frictions. The force ripple was first modeled and compensated using feedforward control. Particle swarm optimization was used to identify the parameters of the Luge model. The proposed method demonstrates excellent modeling accuracy and its effectiveness is experimentally verified in motion control tests.

The third chapter proposes a robust contour control for bi-axial contour following applications. The discrete sliding mode control (DSMC) has been reformulated using the discrete local frame. The proposed control scheme inherits the merits of both contour control and DSMC. The proposed discrete sliding mode contour control (DSMCC) achieves very consistent contour following performances even in the presence of disturbances.

The fourth chapter extends the results of contour control to regular curves in  $\mathbb{R}^3$ . The key is to design and optimize the local moving frame. Most previous researches are limited to bi-axial applications or plane curves. As such, the moving frame method has not been fully studied for regular curves in  $\mathbb{R}^3$ . An optimization method was proposed to solve this problem. The torsion of curves is taken into consideration to optimize the moving frame. Compared to the conventional moving frame (moving trihedrons), the optimized local frame has much smoother transition and therefore eliminates spikes and saturations in control efforts. In addition, an improved contour error estimation method and a chatter free robust inner loop have also been developed to further improve the contour following accuracy.

Chapter 5 discusses our future work of using learning control to further improve the performances of the multi-axis machine tools in mass productions.

## CHAPTER 2      MODELING OF MACHINE TOOLS FOR LOW SPEED MOTIONS

### 2.1    Experimental Setup

In this chapter, the dynamics of one feed drive of the experimental machine tool is identified. The focus of this chapter is to identify the rigid body dynamics as well as the disturbances mainly including force ripple and friction. The machine tool of consideration is a 3-axis micro mill in our lab, see Figure 7. The application of the mill includes the fabrications of high temperature sensors of machinable polymer derived ceramics (PDC) [2] and machinability studies of metal matrix composite (MMC) [49]. All feed drives are equipped with optical encoders. The encoder period of *X-axis* and *Y-axis* are  $4\mu\text{m}$ ; for *Z-axis*, the encoder period is  $18\mu\text{m}$ . Signal multipliers are employed to increase the resolutions to  $0.02\mu\text{m}$  for *X-axis* and *Y-axis*,  $0.09\mu\text{m}$  for *Z-axis*. *X-axis* and *Y-axis* are powered by DRPANIE-015A servo drives from Advanced Motion Control; *Z-axis* is powered by a Soloist CP10 servo drive from Aerotech. All servo drives work in current mode and take standard  $\pm 10\text{Volt}$  control signal from the main controller.

The main controller is a PXIe-8133 from National Instrument (NI). The PXIe controller is equipped with a data acquisition card (DAQ) PXI-7854R receiving quadrature incremental position signals from the signal multipliers. Control signals are sent to servo drives through the on board analog output of PXI-7854R. The block diagram of the control system is illustrated in Figure 8. The identification in this chapter and control algorithms in the later chapters are implemented in Labview Realtime operating system.

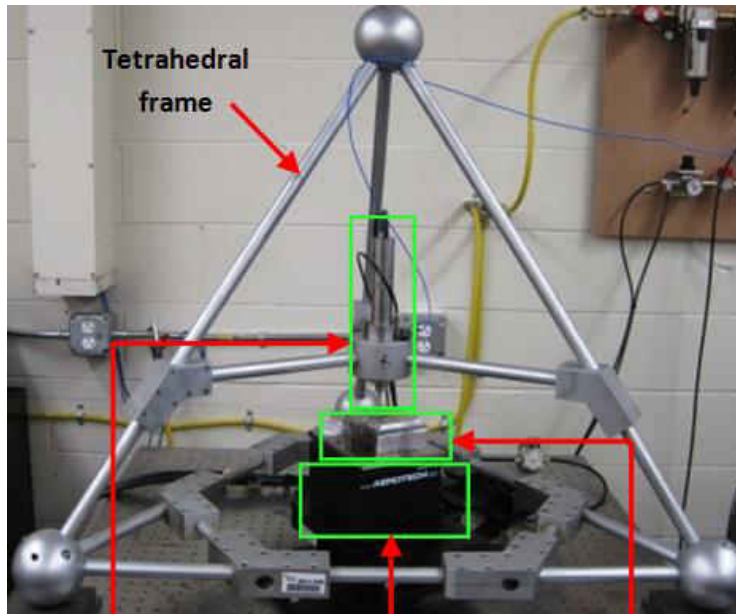


Figure 7: 3-axis micro mill

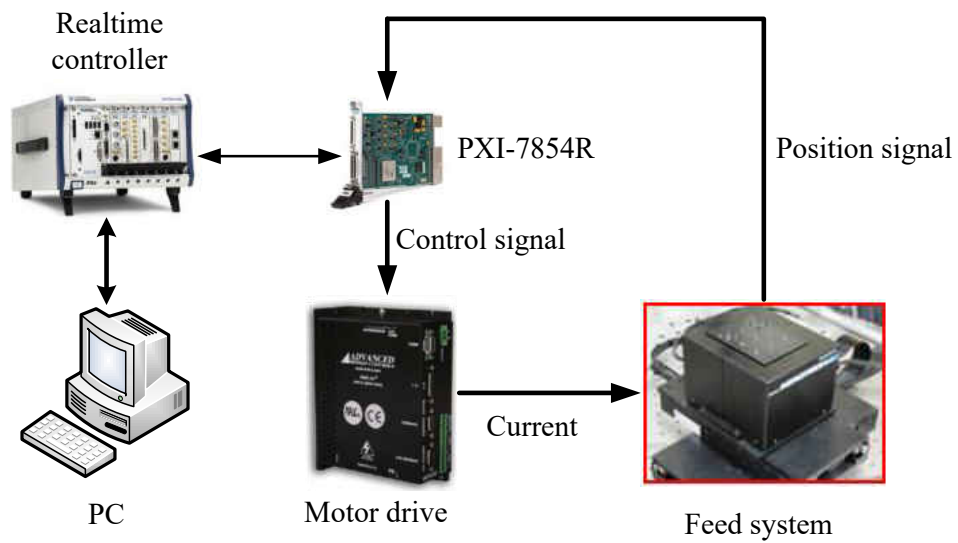


Figure 8: Block diagram of experimental machine tools

To enhance the machining accuracy, all feed drives of the micro-mill are linear-motor-driven. Compared to the conventional lead screw transmission feed systems, linear motors

eliminate the transmission error and reverse backlash [15, 50], which therefore are more capable of precision motions. The significance of obtaining a comprehensive model is that the effectiveness of many control strategies depends on accuracy of plant model, especially those utilizing direct inversion of the system dynamics [11, 51, 52]. For example the performance of ZPETC heavily depends on the fidelity of the plant model [11, 53]. The modeling uncertainties also have direct influences on the stability and convergence of the iterative learning control (ILC) [54-60]. In addition, despite the surge of modern robust control, an accurate and analytical system model always facilitates the control system design. For example, the model of disturbance can be directly implemented as feedforward compensation to provide fast disturbance rejection [23, 61, 62].

For machine tool systems, friction is always a major source of both tracking error and contour error due to its strong nonlinearity and discontinuity. Friction can also lead to many other undesired phenomena like limit cycle and stick-slip motion. In multi-axis contour control, the error caused by friction is referred to as ‘quadrant glitch’ and is illustrated in highlighted red circles in Figure 9.

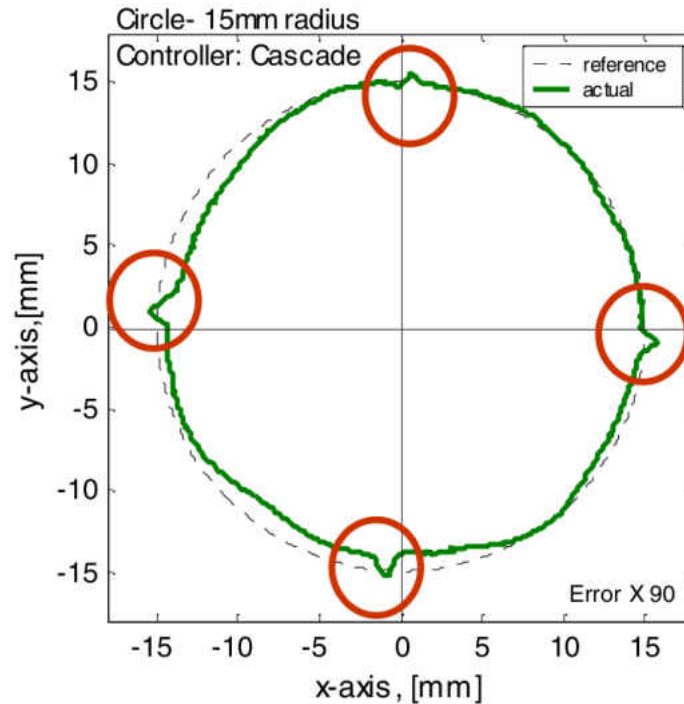


Figure 9: Quadrant glitch caused by frictions [63]

Several examples of friction models are static friction model [64], Gaussian model [65], and LuGre model [66] and general Maxwell model [67]. Simple friction models, such as static friction model, are easy to identify and implement yet cannot always provide satisfactory compensation results. The advanced friction models, for example, the LuGre model, are able to describe many complex behaviors of frictions. Yet the identification of advanced friction models are usually a challenging task [68]. A number of numerical identifications methods have been tested, like Simplex [69] and Monte Carlo process [19, 70]. Yet these methods have the tendency to fall into local optimum. Besides, redundant experiments are needed to provide good initial parameter estimations [71] otherwise these algorithms may not work.

Another factor affecting the accuracy of the friction modeling is the existence of other disturbances. In linear feed drives, force ripples could cause enormous error of friction modeling

and thus must be firstly eliminated. Here we target at solving all the aforementioned problems and establishes a precision model of the feed drive system of the experimental mill. This chapter is organized as the following sections: in section 2, the overall model of the system is developed. In section 3, the model of force ripple is identified and experimentally verified. The force ripple is subdued using model based feedforward compensator. In section 4, the servo gain is identified using a simplified fiction model. In section 5, the classic Lugre model is modified to model frictions of poor symmetry in different directions. However, the additional parameters make the parameter identification of Lugre model a more challenging task. A global numerical optimization method, particle swarm optimization (PSO) [72] is utilized to identify the 8 parameters of the model. In section6, the overall model is tested in both open loop and closed loop tests. The obtained models are analyzed in both time domain and frequency domain. The closed loop tracking experiment also demonstrates that the identified model can greatly enhance the tracking performance. Section 7 concludes the entire chapter.

## 2.2 Dynamics of Experimental Setup

Figure 10 is the dynamics of the feed-drive to be identified.  $k_d$  (*Amp/Volt*) is the ratio between the control signal (*Volt*) and armature current (*Amp*). For the *X-axis* and *Y-axis*,  $k_d$  is 0.35 *Amp/Volt* and 1*Amp/Volt* on the *Z-axis*.  $k_m$  (*Newton/Amp*) represents the force constant,  $d$  denotes lumped disturbances;  $m$  denotes the inertia of the motor mover.

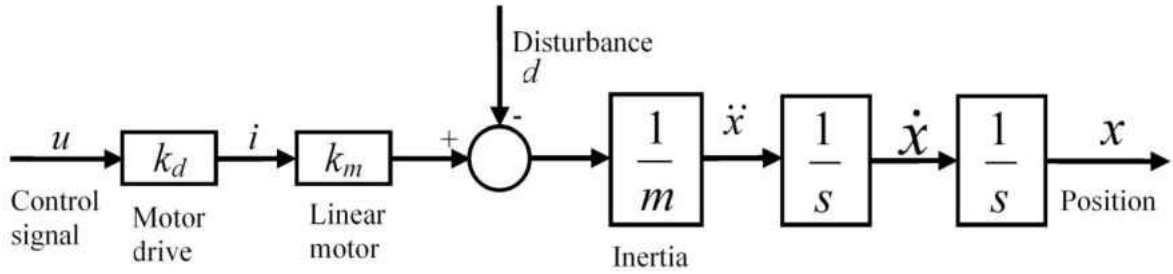


Figure 10: Block diagram of dynamics of direct drive

The block diagram shown in Figure 10 causes identifiability problem since the system inertia is unknown. Therefore the block diagram is modified as shown in Figure 11, where the disturbances act on the control signal  $u$  instead of on the motor mover. In addition, the drive constant  $k_d$ , motor constant  $k_m$  and the inertia  $m$  will be combined to form a new parameter named the ‘equivalent motor constant’ ( $mm/s^2Volt$ ).

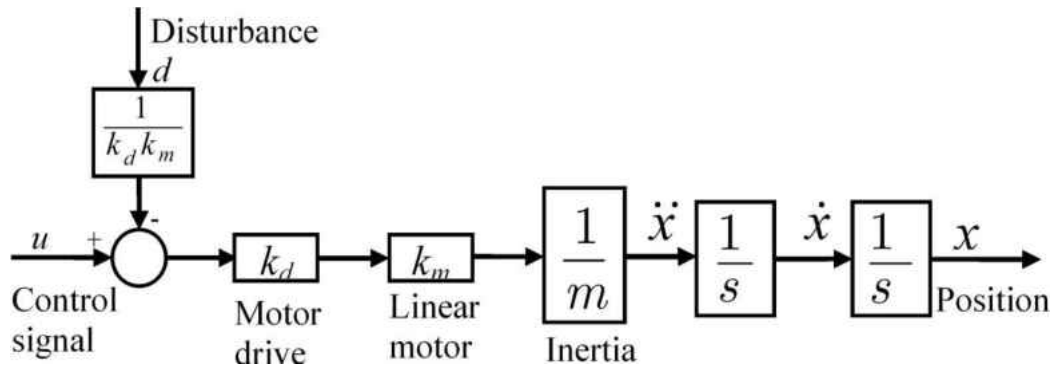


Figure 11: Block diagram of modified dynamics of direct drive

The disturbances  $d$  include both the force ripple and friction

$$d = f^* + f_{ripple}^* \quad (12)$$



where  $f_{ripple}^*$  denotes the force ripple and  $f^*$  is friction. To coincide with Figure 11, the normalized disturbances are

$$\begin{cases} f_{ripple} = f_{ripple}^* (k_d k_m)^{-1} \\ f = f^* (k_d k_m)^{-1} \end{cases} \quad (13)$$

The normalized model of the system is

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{k_d k_m}{m} \end{bmatrix} u + \begin{bmatrix} 0 \\ -\frac{k_d k_m}{m} \end{bmatrix} (f_{ripple} + f) \quad (14)$$

Applying Laplace transform to Eqn (14) we have

$$x(s) = \frac{k_d k_m}{ms^2} (u(s) - f_{ripple}(s) - f(s)) \quad (15)$$

### 2.3 Identification of Force Ripple

Force ripple refers to the position-dependent periodic variation of the force constant  $k_m$  caused by imperfect commutation. The experiment consists of jogging the linear motor at a constant speed (4mm/s) using a proportional-integral (PI) controller [73]. Different load forces are used to impose a tangential force to the linear motor so the influence of mean control signal can be determined. The experimental setup for force ripple test is illustrated in Figure 12. The variations of the control signal are used as a measurement of the force ripple.

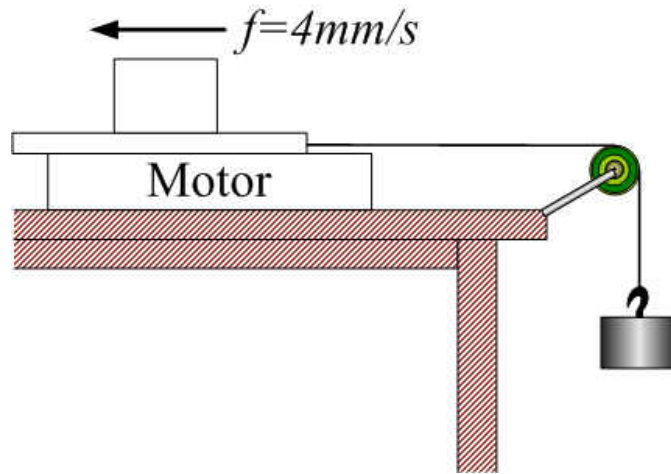


Figure 12: Experiment setup for force ripple test

In each experiment, control signal is sampled after the motion reaches steady state. The control signal is re-sampled in an equal-distance manner, which is similar to the method in [73]. Figure 13, Figure 14 and Figure 15 show different control signals and their mean values. The variations of the control signal clearly prove the existence of the force ripple since the variation of the friction and the load force during a constant speed motion is trivial. Figure 16 shows the spectrum of the control signal in Figure 15.

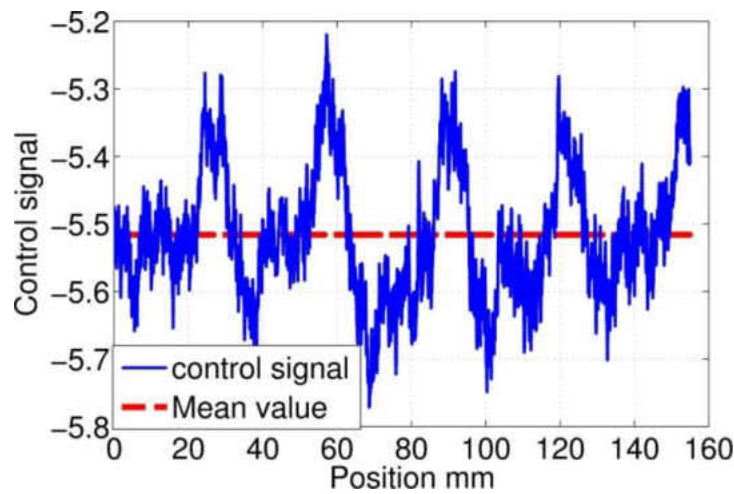


Figure 13: Control signal showing force ripple #1

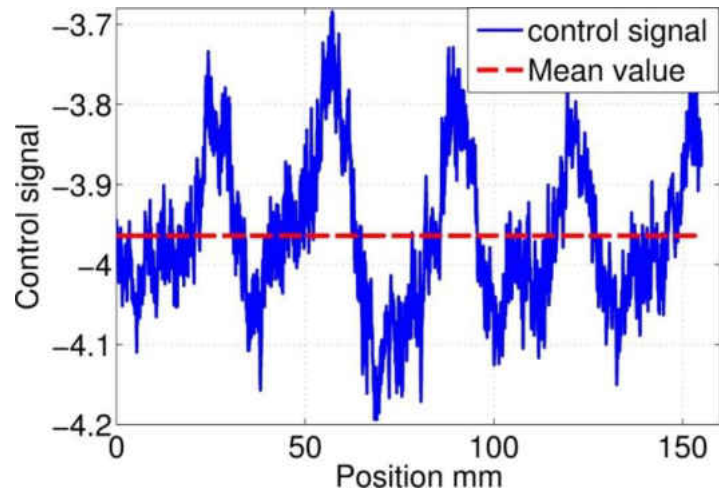


Figure 14: Control signal showing force ripple #2

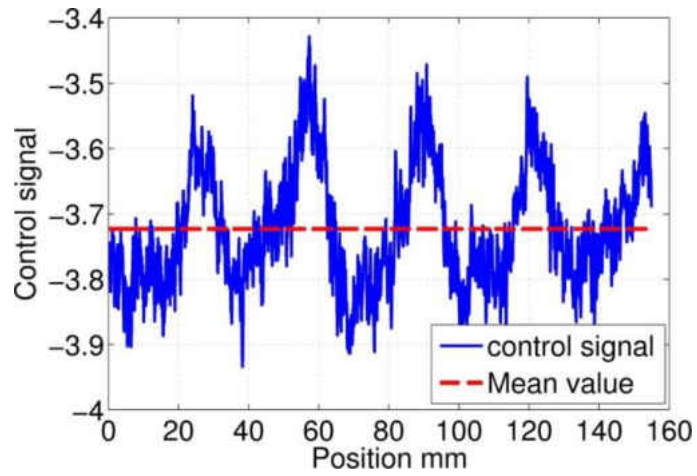


Figure 15: Control signal showing force ripple #3

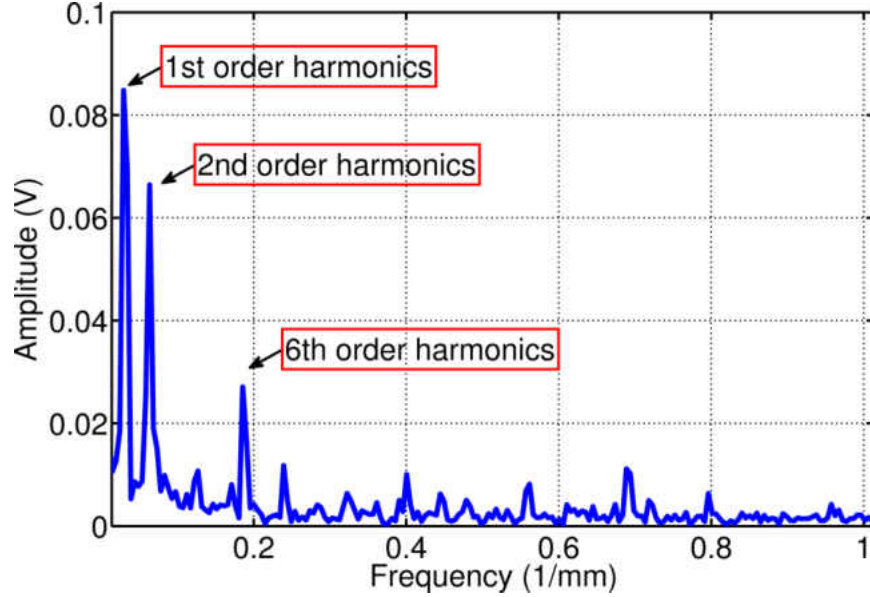


Figure 16: Spectrum of force ripple 3

After analyzing the spectrums of control signals obtained from different experiments, all force ripples are found including the same three major harmonics. The first order harmonic with a period of 0.02931 is determined as a current independent disturbance, which remains almost unchanged in all experiments; the second order harmonic with a period around 0.06351 as well as the sixth order harmonic with a period around 0.1856 are current dependent as their magnitude changes with the imposed loads change. Since the force ripple can be modeled as the summation of individual harmonics, the following function is used to represent the force ripple

$$F(x, u_n) = \sum_{n=1,2,6} g_n(u_{nominal}) \sin\left(\frac{nx}{p} + q_n\right) \quad (16)$$

where  $p$  is the period of the first order harmonica,  $q_n$  is the phase of each harmonics,  $g_n(u_{nominal})$  is the amplitude function of each harmonic and will be determined later,  $u_{nominal}$  is the nominal value of the control signal, which will also be explained in the following

sections. The nonlinear least square method is used to identify the parameters of  $p$  and  $q_n$ . Results are listed in. The applied payloads are not given here since the model of the force ripple does not depend on them.

Table 1: Identification results of force ripples

	No load	Load 1	Load 2	Load 3	Load 4	Load 6
$p$	31.93	31.89	32.00	31.99	31.96	31.94
$g_1$	0.105	0.1	0.11	0.1	0.102	0.96
$q_1$	2.9	2.92	2.96	2.93	2.97	2.91
$g_2$	0.01	0.027	0.036	0.045	0.051	0.066
$q_2$	3.69	3.69	3.79	3.71	3.7	3.63
$g_6$	0.002	0.01	0.015	0.018	0.02	0.029
$q_6$	4.59	4.08	4.35	4.34	4.36	4.23

The final identification results of  $q_n$  and  $p$  are obtained by taking mean values of results from all experiments.  $u_{nominal}$  is the average of control signal over a range of integral multiples of  $p$  so the harmonic components of the force ripple is minimized.  $g_n$  is modeled as a linear function of  $u_{nominal}$

$$g_n(u_n) = a_n u_{nominal} + b_n \quad (17)$$

where  $a_n, b_n$  are

$$\begin{cases} a_1 = 0 \\ b_1 = 0.10 \end{cases}, \begin{cases} a_2 = 0.0105 \\ b_2 = 0.0067 \end{cases}, \begin{cases} a_6 = 0.0049 \\ b_6 = 0.000685 \end{cases}$$

Figure 17 shows the identification results of  $g_2$  and  $g_6$ .

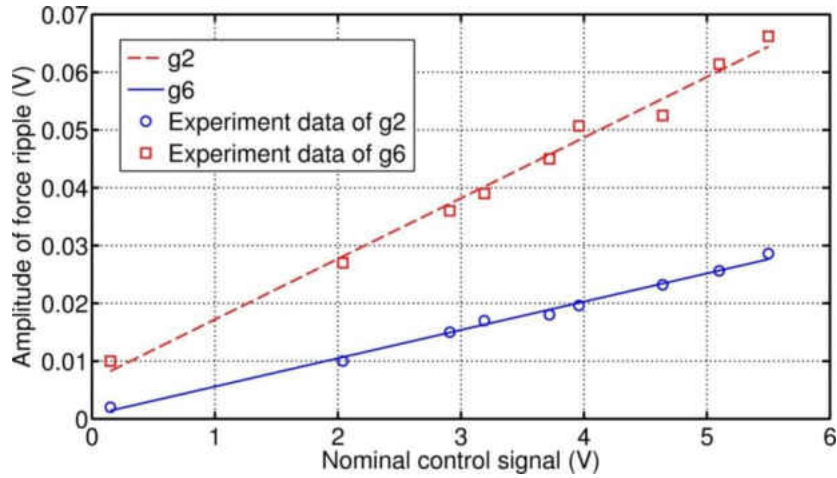


Figure 17: Identification results of  $g_2$  and  $g_6$

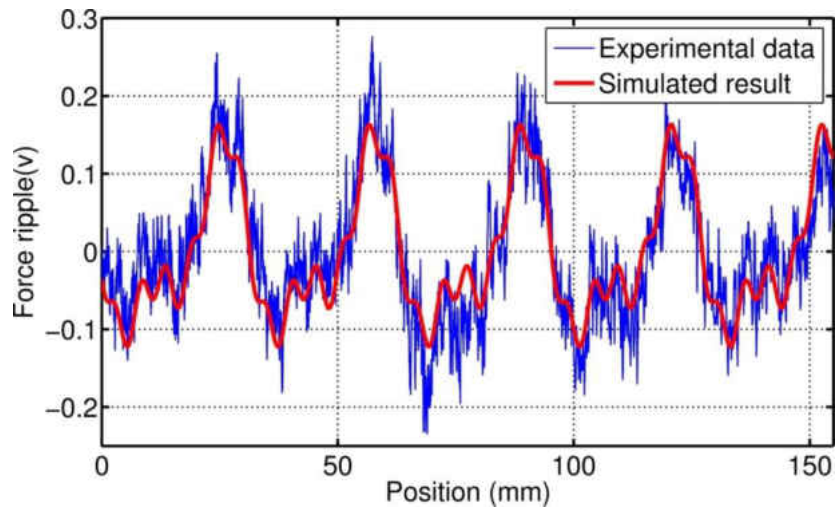


Figure 18: Comparison between actual force ripple and simulated force ripple

Figure 18 is a comparison between the actual force ripple and the model output, which clearly shows a very accurate fit. Experiments described at the beginning of this section are repeated with the identified force ripple model implemented as a feed-forward compensator. Figure 19 shows the block diagram of the control system, where PI is the main feedback control and  $F(x, u)$  is the feed-forward based on the identified model.

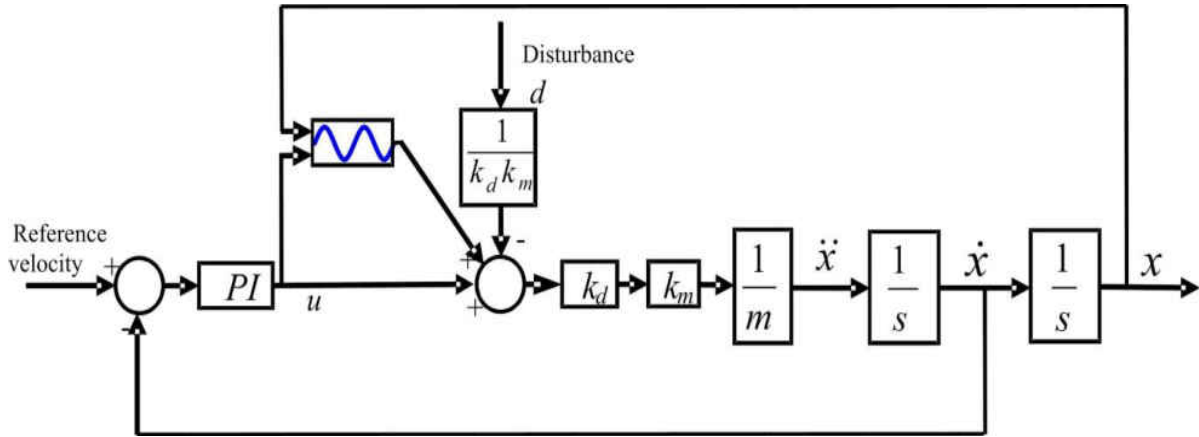


Figure 19: Feedforward control

For comparison purpose, experiments are repeated with feed-forward compensator disabled and enabled, the results of which are shown in Figure 20 and Figure 21, respectively.

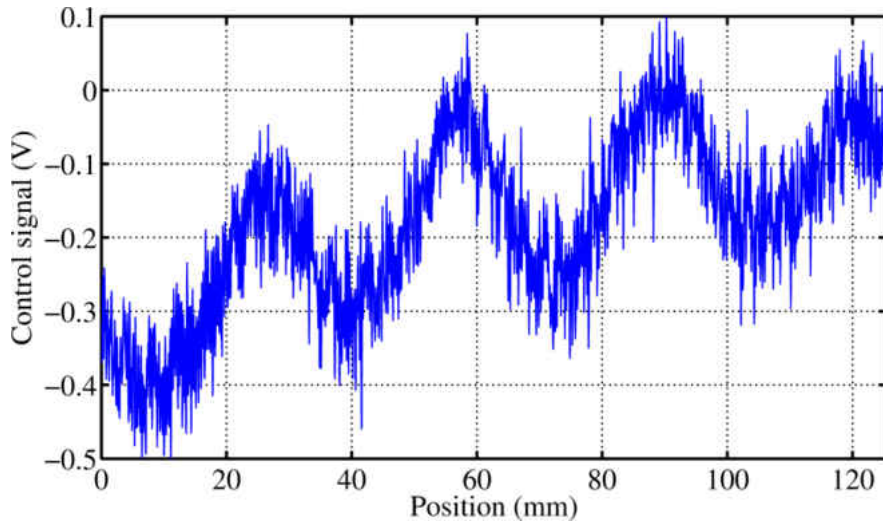


Figure 20: Control signal of PI controller without feedforward compensation

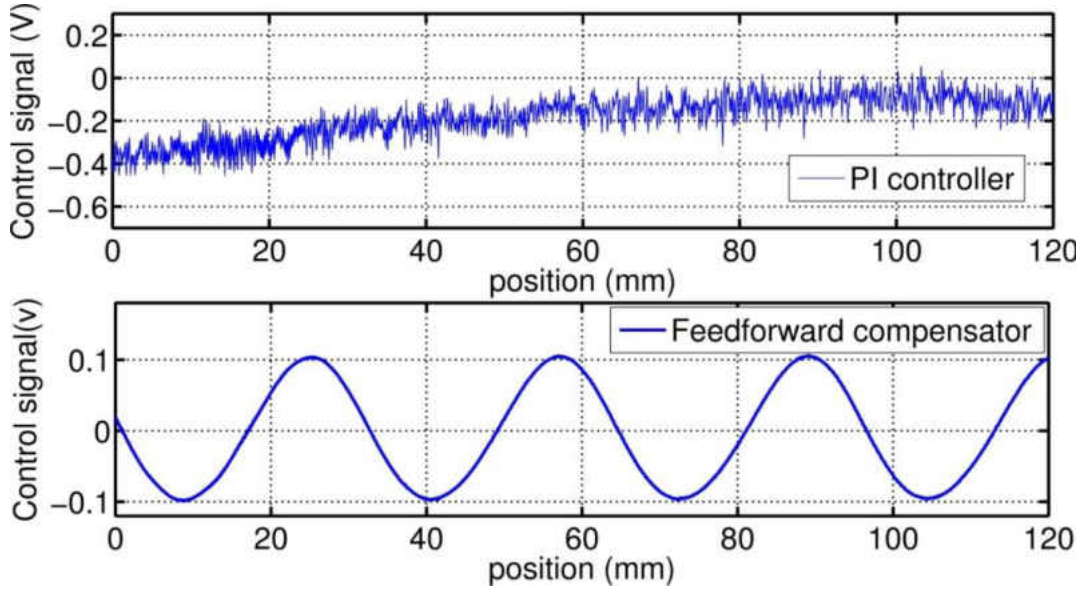


Figure 21: (a) Signal of PI controller (b) Signal of feedforward compensator

From Figure 20, with the feed-forward compensator disabled, the control signal of the PI controller fluctuates due to the force ripple. In Figure 21 (a), the control signal of the PI controller almost becomes much ‘flatter’ with no apparent variation when the compensator is enabled. This clearly demonstrates that the force ripple is successfully modeled and suppressed by the feedforward compensator. Other experiments with different load forces also show similar results, which are omitted for conciseness.

#### 2.4 Identification of Rigid Body Dynamics

The force ripple is modeled and compensated in the previous section

$$f_{ripple} \approx 0 \quad (18)$$

Therefore Eqn (3) can be further simplified to



$$\frac{d^2x}{dt^2} = \frac{k_d k_m}{m} (u - f) \quad (19)$$

and the parameter to be identified is the motor constant

$$k_m = \frac{f_{thrust}}{I} \quad (20)$$

where  $f_{thrust}$  is the thrust produced by the motor and  $I$  is the armature current. However, the motor constant in the manufacturer specifications may be inaccurate since it is measured at stall. In addition, the inertia of motor mover is also unknown. Therefore, instead of identifying all these parameters individually,  $k_d k_m / m$  is identified as one parameter named as the ‘equivalent motor constant’. To avoid the modeling error caused by the nonlinearities of low speed frictions [24], the experiments are conducted using a high speed single-direction motion and the excitation signal is shown in Figure 22.

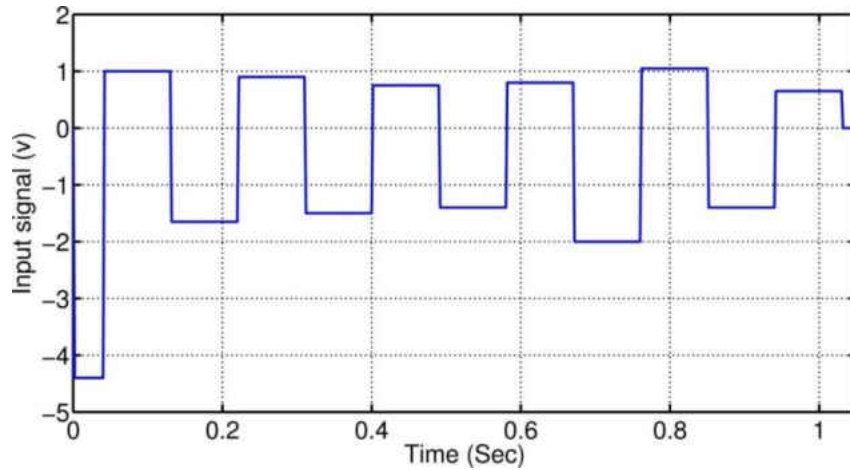


Figure 22: Excitation signal

At high speed, friction can be modeled as [74]

$$f = f_{cou} + vc_{vis} \quad (21)$$

where  $c_{vis}$  is the viscous friction coefficient and  $v$  is the velocity. Substituting Eqn (10) into Eqn (8)

$$\frac{dv}{dt} = \frac{k_d k_m (u - f_{cou} - vc_{vis})}{m} \quad (22)$$

The Eqn (11) in discrete time domain is

$$v(k+1) = p_d v(k) + \frac{k_{vd} k_{md} f_{cou}(k)}{m} + \frac{k_{vd} k_{md} u(k)}{m} \quad (23)$$

where

$$k_{vd} = 1 / (k_d k_m c_{vis} / m) (1 - e^{(-T_s k_d k_m c_{vis} / m)}) \quad (24)$$

$$p_d = e^{(-T_s k_d k_m c_{vis} / m)} \quad (25)$$

$$k_{md} = k_d k_m \quad (26)$$

The velocity in the experiment is negative

$$v(k+1) = p_d v(k) + \frac{k_{md} k_{vd}}{m} c + \frac{k_{vd} k_{md}}{m} u(k) \quad (27)$$

where  $c$  is the coulomb friction, rewrite Eqn (16) in vector form

$$v(k+1) = \begin{bmatrix} v(k) & 1 & u(k) \end{bmatrix} \begin{bmatrix} p_d \\ k_{md} k_{vd} c / m \\ k_{md} k_{vd} / m \end{bmatrix} \quad (28)$$

Applying least square method (LSM) we have

$$Y = (\phi^T \phi)^{-1} \phi^T \theta$$

where  $Y = [p_d \frac{k_{md}k_{vd}c}{m} \frac{k_{md}k_{vd}}{m}]^T$ ,  $\theta = [v(2) \ v(3) \ \dots \ v(n)]^T$ ,  $\phi = \begin{bmatrix} v(1) & 1 & u(1) \\ v(2) & 1 & u(2) \\ \dots & \dots & \dots \\ v(n-1) & 1 & u(n-1) \end{bmatrix}$ , and

the identification results are in Table 2.

Table 2: Identification results of rigid body dynamics

$p_d$	$k_{vd}k_{md}c/m$	$k_{vd}k_{md}/m$
0.998	0.211	0.684

Figure 23 (a) shows the actual velocity response as well as the simulated response. Figure 23 (b) is the velocity error. The identified equivalent motor constant is

$$\frac{k_d k_m}{m} = 342.24 \text{ mm}(s^2 V)^{-1}$$

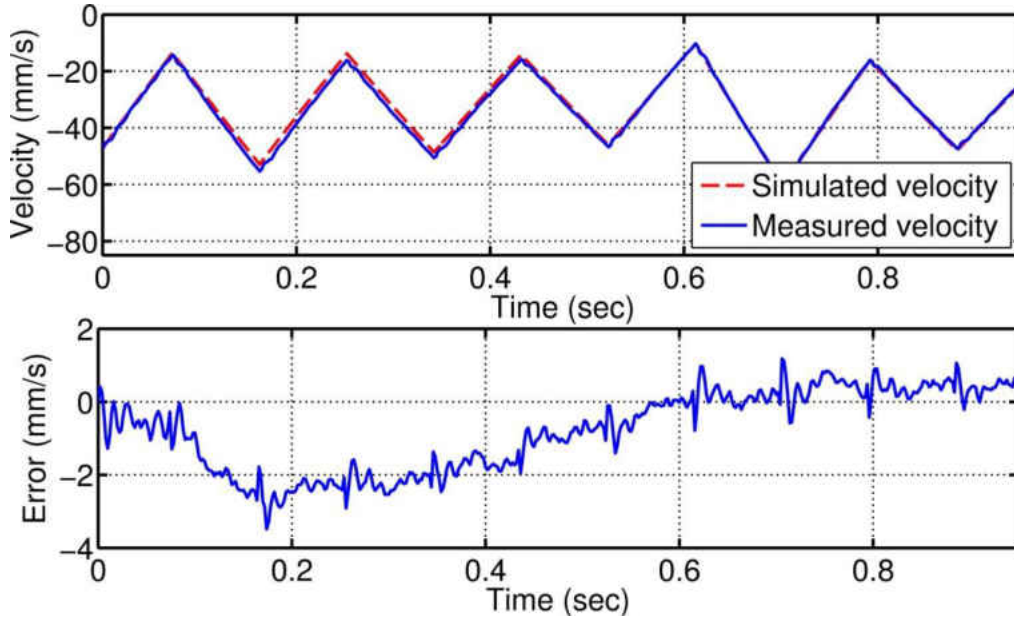


Figure 23: (a) Measured and simulated velocities (b) Error

## 2.5 Identification of Frictions Using Luge Model

The Luge friction model is

$$\begin{cases} F = \sigma_0 z + \sigma_1 \frac{dz}{dt} + a_2 v \\ \frac{dz}{dt} = v - \frac{\sigma_0 |v| z}{g(v)} \\ g(v) = a_0 + a_1 e^{-(v/v_0)^2} \end{cases} \quad (29)$$

where  $z$  is the internal state of the friction,  $a_0, a_1, a_2, v_0$  are static parameters and  $\sigma_0, \sigma_1$  are dynamic parameters.

### 2.5.1 Cost Functions

Since the static parameters and dynamics parameters are identified in two steps, two cost functions are defined, respectively. The four static parameters are identified by static speed-friction mapping measured at various constant-speed motions. By setting

$$\frac{dz}{dt} = 0 \quad (30)$$

we can obtain the model of static friction

$$f_s = a_0 \operatorname{sgn}(v) + a_1 e^{-\left(v/v_0\right)^2} \operatorname{sgn}(v) + a_2 v \quad (31)$$

The cost function for the optimization is

$$g_{static}(a_0, a_1, a_2, v_0) = \sum_{i=1}^n (f_s - f'_s)^2 \quad (32)$$

where  $n$  is the length of data;  $f'_s$  is the friction estimated by (31) and  $f_s$  is the corresponding experimental data.

Once the four static parameters are identified, the simulated response heavily depends on the two dynamic parameters, especially when the system is under small, slowly varying input signals. Thus the goal of optimization is to find the simulated response that matches the experimental measurements. The cost function for optimization of this part is

$$g_{dynamic}(\sigma_0, \sigma_1) = \sum_{i=1}^k (x_i - x'_i)^2 \quad (33)$$

where  $k$  is the data length,  $x_i$  is the simulated response and  $x'_i$  is the experimental response.

### 2.5.2 Particle Swarm Optimization (PSO)

In this section, the identification problems are boiled down to the minimization of the two cost functions. The optimization of static parameters is relatively easy because the cost function is in analytical form which makes it solvable by many optimization methods. On the other hand, for the dynamic parameters optimization, the problem becomes much more difficult since it is almost impossible to find an analytical expression for the cost function of this part, let alone the gradient descent or Hessian matrix. This renders many local search algorithms based on Hessian matrix or gradient descent inapplicable.

Alternative methods to solve this problem include the SIMPLEX and Monte Carlo method. Yet the effectiveness of this technique heavily relies on a good initial guess of the parameters to be identified. Also, the risk of falling into local optima is always a problem of direct search algorithms.

Particle Swarm Optimization (PSO), first reported in [72] is a relatively new global search method inspired by the behaviors of bird flocks. Though the original PSO is designed to deal with the optimization of continuous nonlinear functions, PSO is now capable of a large varieties of optimization problems [75, 76]. The principle of PSO is similar to the genetic algorithm (GA) [77] except that PSO does not have any evolutionary operators like mutation and crossover.

PSO starts working with a group of random solutions with each solution called a particle. Every particle has a memory unit to store its personal best (pbest) and its corresponding fitness. The best value of the pbest of the entire particle swarm is called global best (gbest), which is shared among all the swarm. In this manner, each particle is aware of both the personal best and

the global best. Using this information, particles adapt velocities toward the possible best positions, which provide a mechanism of global convergence. Normally particles are only allowed to search within a predefined searching area. If a particle hits the boundary, it would be bounced back into the searching area.

Using PSO with  $M$  particles to solve an  $N$ -dimensional optimization problem,  $i$  is denoted as the iteration number; the  $k_{th}$  particle in the swarm is represented by a  $N$ -dimensional vector

$$\mathbf{X}_k = [X_{k1} X_{k2} \dots X_{kN}] \quad (34)$$

The velocity vector and updated equation are [78]

$$\mathbf{V}_k(i+1) = w\mathbf{V}_k(i) + c_1 r_1 (\mathbf{pbest}_k - \mathbf{X}_k(i)) + c_2 r_2 (\mathbf{gbest} - \mathbf{X}_k(i)) \quad (35)$$

$$\mathbf{X}_k(i+1) = \mathbf{X}_k + \mathbf{V}_k(i+1) \quad (36)$$

where  $w$  represents inertia,  $c_1$  is cognitive acceleration constant;  $c_2$  is the social acceleration constant;  $r_1$  and  $r_2$  are two random numbers uniformly distributed between 0 and 1.

### 2.5.3 Parameter Identification

To establish the static speed-friction mapping, constant speed experiments are conducted at 20 different speeds ( $-25\text{mm/s} \sim 25\text{mm/s}$ ). In order to take into account of the difference between the frictions in the negative direction and in the positive direction, Eqn (31) is modified to

$$f_s = a_0 \operatorname{sgn}(v) + a_1 e^{-\left(\frac{v}{v_0}\right)^2} \operatorname{sgn}(v) + a_2 v - \operatorname{sign}(v) a_3 \quad (37)$$

where

$$\operatorname{sign}(v) = \begin{cases} 1, & \text{if } v < 0 \\ 0, & \text{otherwise} \end{cases}$$

and  $a_3$  is a dummy parameter representing the average of the difference between the bidirectional frictions. Optimization of static parameters becomes:

$$\text{Minimization for } g_{static}(a_0, a_1, a_2, a_3, v_0)$$

$$\text{Subject to: } a_0, a_1, a_2, a_3, v_0 \in [0, \infty]$$

The swarm size is 20 and the max iteration number is 200, Figure 24 shows the cost function decreasing as the iteration number increases. Figure 25 is the comparison between the identified model and the friction data obtained experimentally. The identified parameters are listed in Table 3.

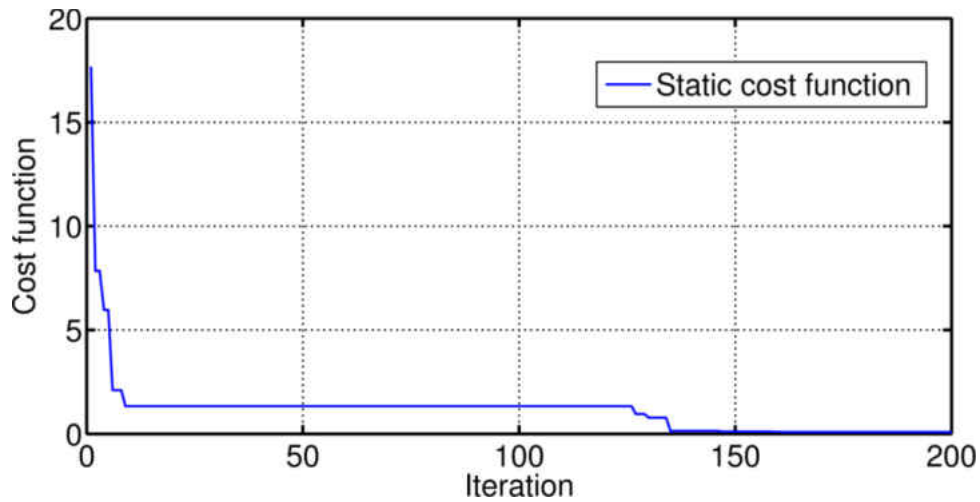


Figure 24: Convergence of cost function of static parameters



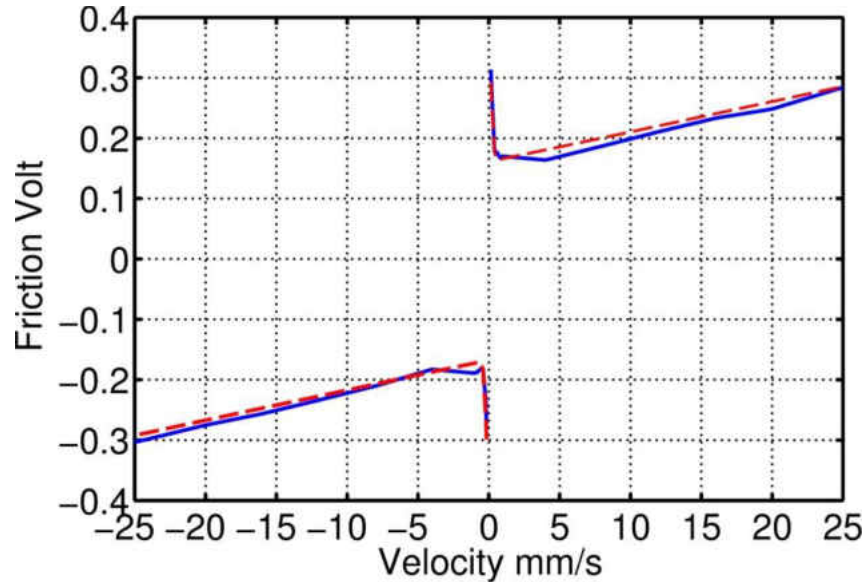


Figure 25: Static speed-friction mapping and results of identification

Table 3: Identification results of static parameters

$a_0$	$a_1$	$a_2$	$a_3$
0.16	0.19	0.005	0.0066

During the stick-slip motion with frequent zero velocity crossings, the simulated motion is sensitive to the variations of  $\sigma\theta$  and  $\sigma l$ . In addition, it is also found that during such motion the response is even more sensitive to the differences between the frictions in the negative direction and in the positive direction.

Due to the existence of the sign function, direct implementation of Eqn (37) occasionally causes unwanted chatters. Thus the Lugre model (29) is modified to

$$\begin{cases} F = \sigma_0 z + \sigma_1 \frac{dz}{dt} + a_2 v - \text{sat}(v) a_3 \\ \frac{dz}{dt} = v - \sigma_0 |v| z g(v)^{-1} \\ g(v) = a_0 + a_1 e^{-(v/v_0)^2} \end{cases} \quad (38)$$

where

$$\text{sgn}(v) = \begin{cases} -v / v_b, & \text{if } v_b > v > 0 \\ -1, & \text{if } v \geq v_b \\ 0, & \text{otherwise} \end{cases}$$

and  $a_3'$  represents the average difference between the frictions in different directions and  $v_b$  is a parameter that can be freely chosen. In this part of experiment,  $v_b$  is  $0.05\text{mm/s}$ . To induce maximum stick-slip motion, a small and slowly varying sinusoid as shown in Figure 26 is used as the excitation signal.

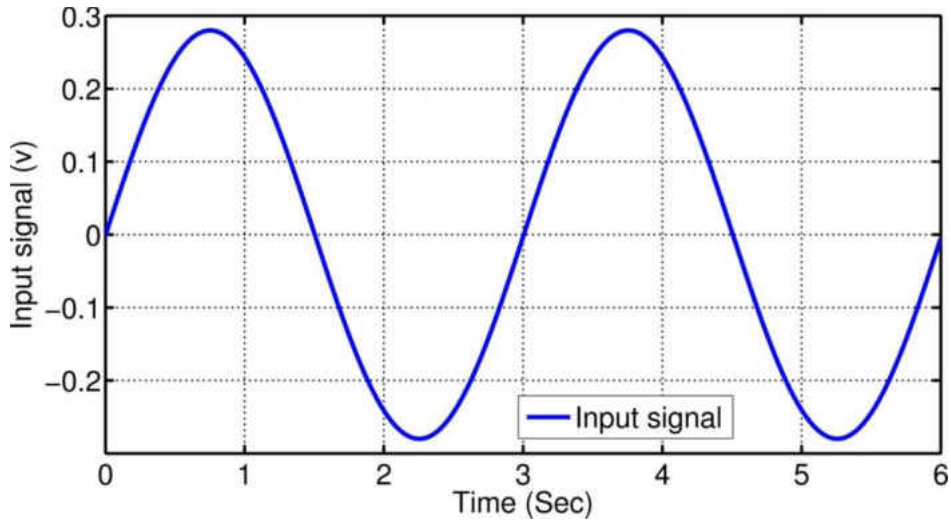


Figure 26: Excitation signal for dynamic parameter identification

The optimization of the dynamic parameters becomes

Minimization of  $g_{dynamic}(\sigma_0, \sigma_1, a'_3)$ , subject to:  $\sigma_0, \sigma_1, a'_3 \in [0, \infty]$

The swarm size is 20 and Figure 27 shows the cost function decreasing as the iteration increases and the optimized parameters are listed in Table 4.

Table 4: Identification results of dynamics parameters

$\sigma_0$	$\sigma_1$	$a'_3$
6.104	0.0592	0.0107

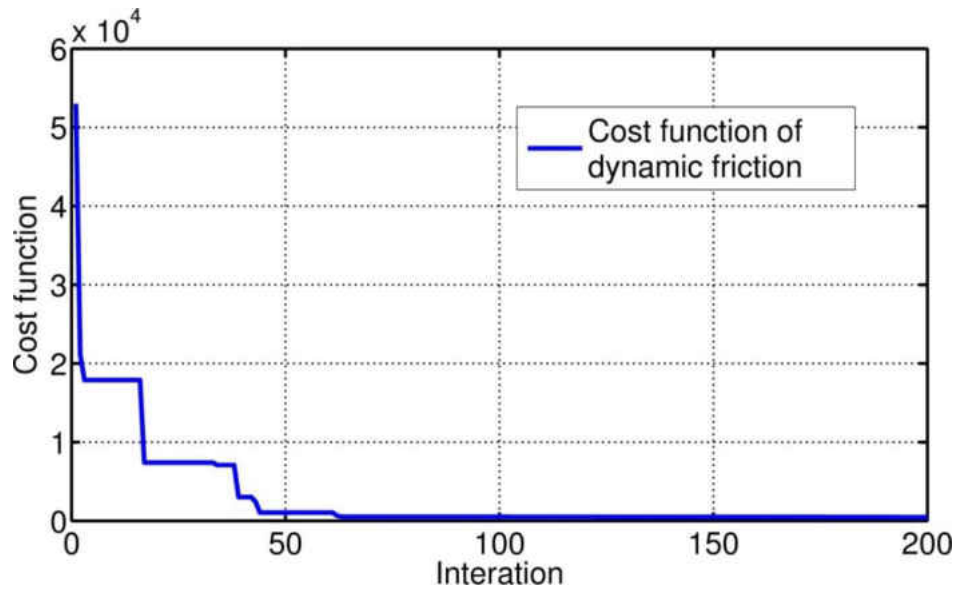


Figure 27: Convergence of cost function of dynamic parameters

Figure 28 is the comparison between the experimental data and simulated response, showing that the two responses are sufficiently close. It is clear that, the proposed method successfully capture the characterizations of frictions during low speed motions.

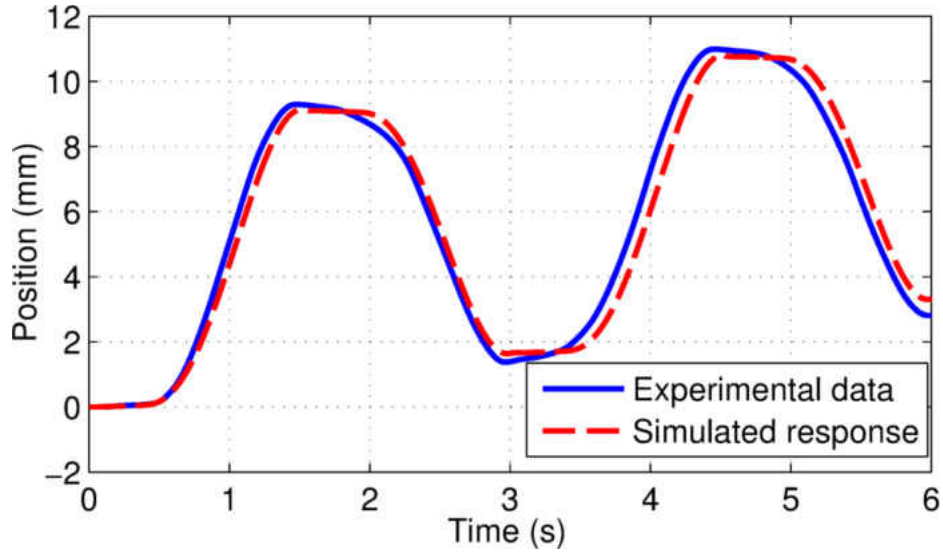


Figure 28: Experimental and simulated responses

## 2.6 Experimental Validations

In this section, the models identified in previous sections are tested experimentally. The first test utilizes the method proposed in [79] to test the closed loop responses of the model. The second experiment is aimed at improving the performance of a pole placement controller using the identified model.

### 2.6.1 Closed Loop Test

In this test, the models of friction and force ripple are implemented as feed-forward compensation. If both friction and force ripple perfectly canceled out, the transfer function of the compensated system is the double integrator

$$g(s) = \frac{342.238}{s^2} \quad (39)$$

The magnitude frequency response of Eqn (36) is

$$M(\omega) = \left| \frac{342.238}{\omega^2} \right| \quad (40)$$

A proportional controller is used to close the feedback loop

$$u_p = PE \quad (41)$$

where  $E$  is the position error and  $P$  is the proportional gain. The closed loop transfer function under the proportional control becomes

$$g_p(s) = \frac{342.238p}{s^2 + 342.238p} \quad (42)$$

The closed loop transfer function has two pure imaginary poles at  $\pm(342.238P)^{0.5}j$  and its steady state response to a step input is a constant oscillation. In this experiment, even a small modeling error will accumulate over time and soon cause large prediction error for the oscillation that will soon become either unstable if the prediction of model is larger than real system (energy is constantly added) or damped rapidly if the model prediction is insufficient (energy is continually consumed). Figure 29 shows the experimental results and the simulated responses with a step input of  $-4mm$ , which show a very good match between the two.

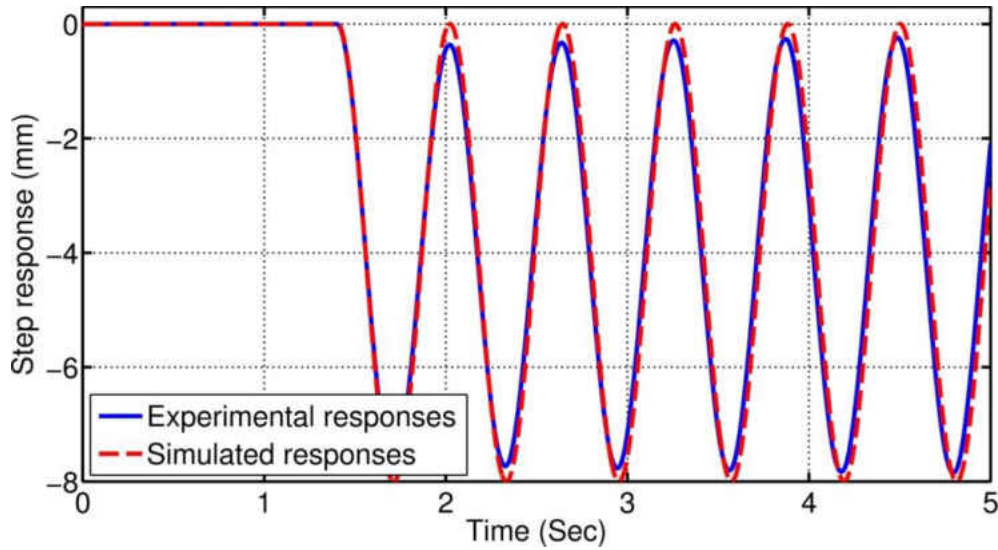


Figure 29: Closed loop step response

### 2.6.2 Tracking Test

This experiment is conducted to show the tracking performance improvement after implementing the identified model. A pole placement controller (PPC) is designed by placing a pair of dominant poles at  $131.95 \pm 134.61j$ . The reference command is a  $0.4\text{Hz}$  sinusoidal signal of a magnitude of  $5\text{mm}$ . The frequency of reference signal is sufficiently lower than the bandwidth of the controller. To demonstrate the performance improvements, the feed-forward compensation is disabled in the first 25 seconds and is enabled in the second 25 seconds. Figure 30 shows the tracking error, which is very large when the feed-forward compensation is disabled. The tracking error is significantly reduced after the feed-forward compensation is enabled, which clearly shows its effectiveness.

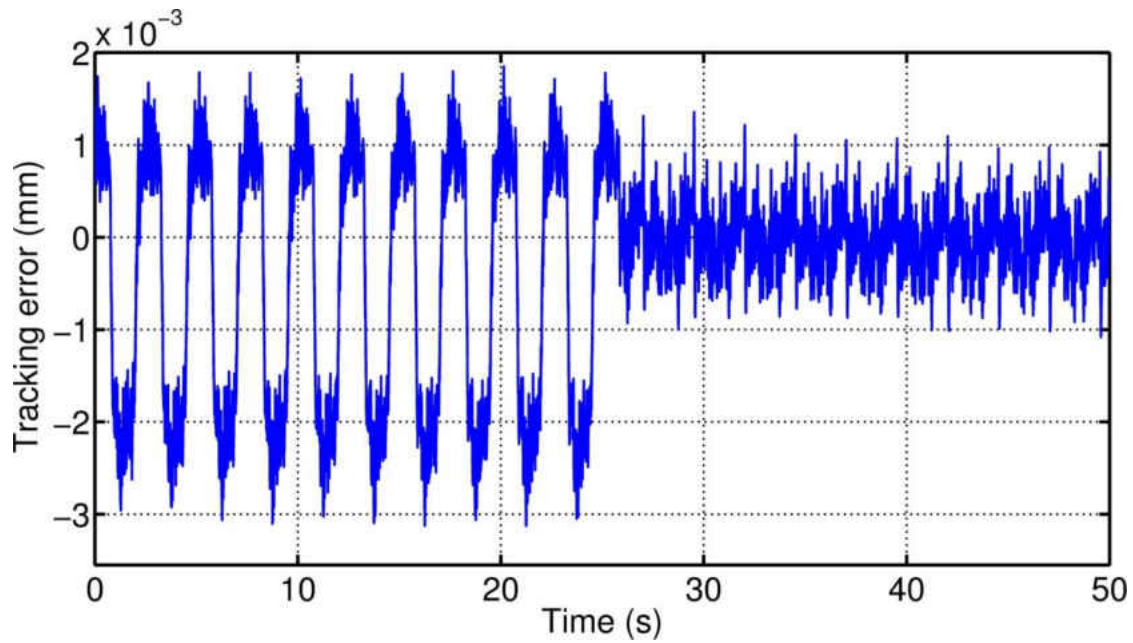


Figure 30: Tracking error comparison

Figure 31 displays the control signal of the PPC. In the section without feed-forward compensation, the wavy control signal of PPC clearly shows the disturbances caused by the friction and force ripple. When the feed-forward compensation is enabled, the control signal becomes a zero mean noise-like signal, proving that both friction and force ripple are eliminated by the feed-forward compensation.

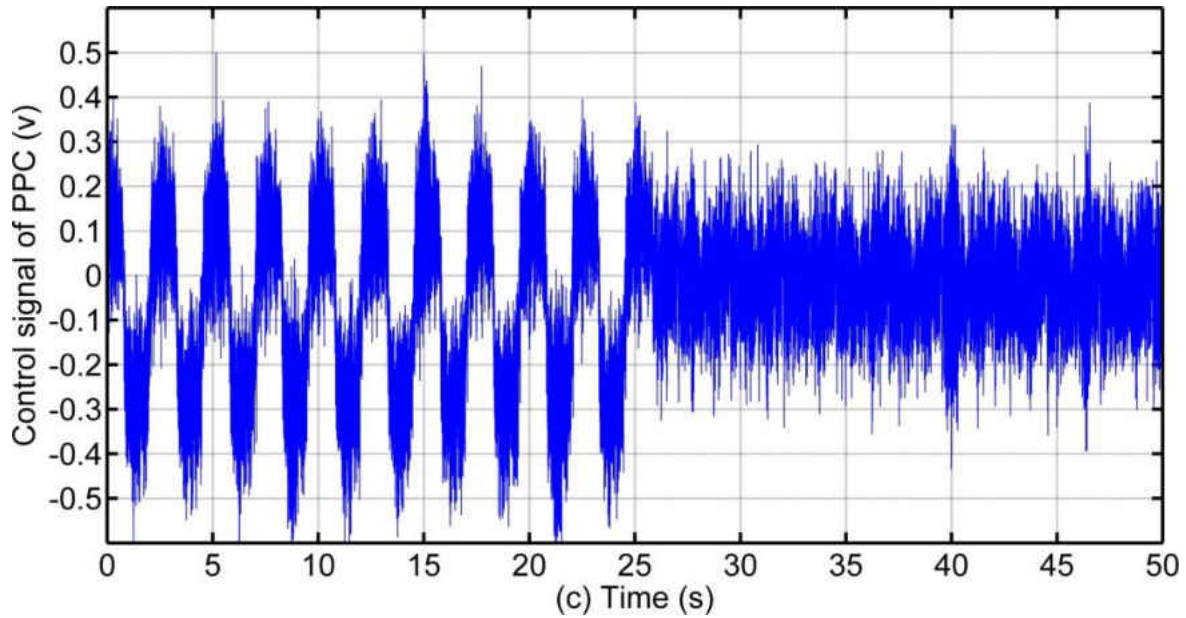


Figure 31: Control signal comparison

## 2.7 Summary

In this chapter, a systematic modeling method of the direct drive system is proposed. Friction and force ripple are decoupled and individually modeled. The effectiveness of the mode is implemented as a feed-forward compensator and its effectiveness is demonstrated experimentally. The proposed modeling method can be readily applied to other DDFs for better performances.



## CHAPTER 3      ROBUST CONTROL FOR PLANAR CONTOUR FOLLOWING

A systematic modeling method has been developed in chapter 2, in this chapter we aim at a new control scheme that combines the robustness of discrete sliding mode control (DSMC) and the contour control using the tangent line approximation. The proposed system reformulates DSMC to achieve consistent the contour following performance even in the presence of disturbances.

### 3.1 Discrete Sliding Mode Control (DSMC)

In industrial environments, mechanical systems are always subject to a certain level of uncertainties or disturbances. These disturbances lead to performance degradation if not carefully addressed. One way to deal with this problem is to accurately model and compensate the disturbance like work done in chapter 2. However, comprehensive modeling could be time consuming and the model fidelity might be lost over time. DSMC provides an alternative solution to deal with this problem. By driving the system trajectory onto a carefully selected hyper-plane, DSMC is able to achieve excellent performance despite of the uncertainties and disturbances. DSMC also features several unique advantages lacked in the well-established continuous SMC [80-82]. However, these researches are solely for performance improvement of single axis and are not optimized for multi-axis applications. We first briefly introduce the design of the conventional DSMC before proposing the robust contour controller.

DSMC is based on the concept of discrete sliding surface. A widely used sliding surface is

$$s(k) = C_s x_s(k) \quad (43)$$

where  $C_s$  is the coefficient vector of proper dimensions and  $x_s(k)$  is the state variable vector. Eqn (43) also represents the distance between the system trajectory and the ideal sliding surface. The essence of DSMC is to forces the system trajectory to approach and subsequently stay on the ideal sliding surface, i.e.

$$s(k) = 0 \quad (44)$$

which is the well-known ideal quasi-sliding mode (IQSM) [83]. It should be noted that due to factors like disturbance, finite sampling rate, IQSM is practically impossible. A more practical goal is to design a controller confining the system trajectory within a narrow band around the IQSM, or the so called quasi sliding mode band (QSMB)

$$-\Delta < s(k) < +\Delta \quad (45)$$

where  $2\Delta$  is the width of the band. A more explicit guideline for the DSMC design is the reachability condition [84]

$$|s(k+1)| < |s(k)| \quad (46)$$

Among the various strategies satisfying the reachability condition, a very effective one is the reaching law [81]. The discrete-time domain reaching law is obtained by applying Euler's forward difference to the continuous version of reaching law, i.e.

$$\frac{s(k+1) - s(k)}{\sigma} = -\eta \text{sign}(s(k)) - qs(k) \quad (47)$$

where  $\eta$  is a positive constant called switching gain and  $q$  is a constant satisfying  $0 < (1 - q\sigma) < 1$ .

*Remark 1:* With a stable IQSM, DSMC system designed using the reaching law approach is always stable [81, 83].

For  $|s(k)| > \eta(1 - q\sigma)^{-1}$ , reaching law (47) satisfies the reachability condition (46).

The switching boundary is

$$|s(k)| < \eta\sigma(1 - q\sigma)^{-1} \quad (48)$$

where the system trajectory crosses the ideal sliding surface  $s(k) = 0$  at the next sampling time. After entering the switching boundary (48), the system trajectory will remain inside a smaller boundary layer

$$|s(k)| < \frac{\eta\sigma}{1 - q} \quad (49)$$

Eqn (49) stands for the idea quasi-sliding mode band (QSMB) [83]. System robustness can be enhanced by increasing the switching gain. On the other hand, from Eqn (49) it is clear that larger switching gain results in a wider QSMB and hence poorer steady state accuracy. Plus the switching control may excite dynamics and even cause chatters. Since the unmolded dynamics is mostly of high frequency, the chatter problem can be alleviated by using a low pass filter to smooth out the switching control.

### 3.2 Discrete Sliding Mode Contour Control (DSMCC) for Nominal System

The dynamics for each feed drive, including the disturbances, can be written as

$$q^i(s) = g^i(s)u^i(s) + d^i(s) \quad (50)$$

where the superscript  $i = X, Y$  stands for  $X$ -axis and  $Y$ -axis, respectively.  $q^i(s)$  is position,  $u^i(s)$  is control effort and  $d^i(s)$  is the lumped disturbances including external disturbances and modeling uncertainties.  $g^i(s)$  is the nominal model of the feed drive of the following double integrator structure

$$g^i(s) = \frac{\beta^i}{s^2} \quad (51)$$

where  $\beta^i$  is the controller gain. Choosing position and velocity as the state variables, i.e,  $x^i(k) = [x_p^i(k) \ x_v^i(k)]^T$ , we can obtain the second order approximation of system (51) in the discrete time domain (zero-order-hold equivalent)

$$\begin{aligned} x^i(k+1) &= Ax^i(k) + B^i u_i(k) \\ q^i(k) &= Cx^i(k) \end{aligned} \quad (52)$$

where the state vector is  $A = \begin{bmatrix} 1 & \sigma \\ 0 & 1 \end{bmatrix}$  and  $\sigma$  is the sampling time,  $C = [1 \ 0]$  and  $B^i = \begin{bmatrix} b^i \sigma / 2 \\ b^i \end{bmatrix}$ .

Model (52) is preferred for its simplicity and the only unknown parameter  $b^i$  can be readily identified. We start our design using model (52) and address the disturbance problem later on.

Denote the reference trajectory vector as  $R^i(k) = [R_p^i(k) \ R_v^i(k)]^T$  and the tracking error vector as

$$x_e^i = [x_{pe}^i \ x_{ve}^i]^T$$

$$x_e^i(k) = R^i(k) - x^i(k) \quad (53)$$

Substituting Eqn (53) into Eqn (52) we have

$$x_e^i(k+1) = Ax_e^i(k) + R^i(k+1) - AR^i(k) - B^i u^i(k) \quad (54)$$

Denote the desired acceleration as  $R_a^i(k)$ , the second order approximation of the desired trajectory  $R_p^i(k)$  is given by

$$R^i(k+1) - AR^i(k) = \begin{bmatrix} R_p^i(k+1) - R_p^i(k) - \sigma R_v^i(k) \\ R_v^i(k+1) - R_v^i(k) \end{bmatrix} = \begin{bmatrix} 0.5\sigma^2 R_a^i(k) \\ \sigma R_a^i(k) \end{bmatrix} \quad (55)$$

Substituting Eqn (55) into Eqn (54) we can obtain

$$x_e^i(k+1) = Ax_e^i(k) + \begin{bmatrix} 0.5\sigma^2 R_a^i(k) \\ \sigma R_a^i(k) \end{bmatrix} - B^i u^i(k) \quad (56)$$

The proposed DSMCC consists of a linear time varying (LTV) feedback control and a linear time invariant (LTI) feed-forward control, i.e.,

$$x_e^i(k+1) = Ax_e^i(k) + \begin{bmatrix} 0.5\sigma^2 R_a^i(k) \\ \sigma R_a^i(k) \end{bmatrix} - B^i (u_{FB}^i(k) + u_{FF}^i(k)) \quad (57)$$

The feedforward control is designed as

$$u_{FF}^i(k) = \sigma(b^i)^{-1} R_a^i(k) \quad (58)$$

Substituting Eqn (58) into Eqn (57) we have

$$\begin{cases} x_e^i(k+1) = Ax_e^i(k) - B^i u_{FB}^i(k) \\ e^i(k) = Cx_e^i(k) \end{cases} \quad (59)$$

, which is the state space representation of the error dynamics. Next we proceed to design  $u_{FB}$ , starting with the task coordinate in the discrete-time domain. If the desired contour is given in the form of quintic spline, then for the  $k$ th reference points  $D_k$

$$\dot{t}(k) = \frac{x'(k)e_x + y'(k)e_y + 0e_z}{\|x'(k)e_x + y'(k)e_y + 0e_z\|} \quad (60)$$

Further define the normal unit vector

$$\vec{n}(k) = \vec{b}(k) \times \vec{t}(k) \quad (61)$$

where  $\vec{b}(k) = [0 \ 0 \ 1]^T$  is the unit vector normal to the machining plane. The last elements of  $\vec{t}(k)$  and  $\vec{n}(k)$  are both dropped since they are trivial for biaxial applications. A Cartesian local frame can be established using  $\vec{t}(k)$  and  $\vec{n}(k)$  as basis vectors. The transformation between the machine frame  $XY$  and the local frame  $tn$  is

$$\mathbf{T}(k) = \begin{bmatrix} \vec{t}(k) & \vec{n}(k) \end{bmatrix}^{-1} \quad (62)$$

where  $\mathbf{T}(k) \in \mathbb{R}^{2 \times 2}$ .  $\mathbf{T}(k)$  is unitary, i.e.,

$$\mathbf{T}(k)^{-1} = \mathbf{T}(k)^T$$

The tracking error referred to the task coordinate is

$$\varepsilon(k) = \begin{pmatrix} \varepsilon_t(k) \\ \varepsilon_n(k) \end{pmatrix} = \mathbf{T}(k) \begin{bmatrix} x_{pe}^X(k) \\ x_{pe}^Y(k) \end{bmatrix} \quad (63)$$

where  $\varepsilon_t(k)$  is the tangent error and  $\varepsilon_n(k)$  is the estimation of the contour error. Define the following state variables for the contour controller design

$$\zeta(k)_{4 \times 1} = \begin{bmatrix} \zeta_1(k) \\ \zeta_2(k) \\ \zeta_3(k) \\ \zeta_4(k) \end{bmatrix} = \bar{\mathbf{T}}(k) \begin{bmatrix} x_{pe}^X(k) \\ x_{pe}^Y(k) \\ x_{ve}^X(k) \\ x_{ve}^Y(k) \end{bmatrix} \quad (64)$$

and  $\bar{\mathbf{T}}(k)$  is given by

$$\bar{\mathbf{T}}(k) = \begin{bmatrix} \mathbf{T}(k) & 0_2 \\ 0_2 & \mathbf{T}(k) \end{bmatrix}$$

where  $0_2$  is a two by two zero matrix. From the construction of  $\zeta(k)$ ,  $\zeta_2(k)$  is the estimation of the contour error and  $\zeta_1(k)$  is the tangent error component. Using transformation (64) to reformulate Eqn (59)

$$\zeta(k+1) = \bar{\mathbf{T}}(k+1)\bar{A}\bar{\mathbf{T}}(k)^{-1}\zeta(k) - \bar{\mathbf{T}}(k+1)\bar{B}u_{FB} \quad (65)$$

$u_{FB} = \begin{bmatrix} u_{FB}^X & u_{FB}^Y \end{bmatrix}^T$ ,  $\bar{A}$  is the extension of the state transition matrix  $A$

$$\bar{A} = \begin{bmatrix} I_2 & \sigma I_2 \\ 0_2 & I_2 \end{bmatrix},$$

$\bar{B}$  is

$$\bar{B} = \begin{bmatrix} B_2\sigma/2 \\ B_2 \end{bmatrix} \quad (66)$$

where  $B_2 = \begin{bmatrix} b^X & 0 \\ 0 & b^Y \end{bmatrix}$  and  $b^X, b^Y$  are the controller gains of X-axis and Y-axis, respectively.

$B_2$  is reversible from the controllability of the system. We further define the following transformation

$$\phi(k)_{4 \times 1} \triangleq \begin{bmatrix} \phi_h(k)_{2 \times 1} \\ \phi_l(k)_{2 \times 1} \end{bmatrix} = \mathbf{F}^{-1} \zeta(k) \quad (67)$$

where  $\mathbf{F} = \begin{bmatrix} I_2 & 0.5\sigma I_2 \\ 0_2 & I_2 \end{bmatrix}$ . Using the state transformation (67), Eqn (65) becomes

$$\phi(k+1) = \mathbf{F}^{-1} \bar{\mathbf{T}}(k+1) \bar{\mathbf{A}}(\bar{\mathbf{T}}(k))^{-1} \mathbf{F} \phi(k) - \mathbf{F}^{-1} \bar{\mathbf{T}}(k+1) \bar{\mathbf{B}} u_{FB}$$

$\mathbf{F}^{-1} \bar{\mathbf{T}}(k+1)$  and  $\bar{\mathbf{T}}(k)^{-1} \mathbf{F}$  commute, therefore

$$\phi(k+1) = \bar{\mathbf{T}}(k+1) \mathbf{F}^{-1} \bar{\mathbf{A}} \mathbf{F} (\bar{\mathbf{T}}(k))^{-1} \phi(k) - \bar{\mathbf{T}}(k+1) \mathbf{F}^{-1} \bar{\mathbf{B}} u_{FB} \quad (68)$$

Eqn (68) can be further simplified to

$$\phi(k+1) = \bar{A}_w(k) \phi(k) - \bar{\mathbf{T}}(k+1) \bar{B}_w u_{FB} \quad (69)$$

where  $\bar{B}_w = \mathbf{F}^{-1} \bar{\mathbf{B}} = \begin{bmatrix} 0_2 \\ B_2 \end{bmatrix}$  and the state transition matrix  $\bar{A}_w(k)$  is given by

$$\bar{A}_w(k) = \begin{bmatrix} \tilde{\mathbf{T}}(k) & \sigma \tilde{\mathbf{T}}(k) \\ 0_2 & \tilde{\mathbf{T}}(k) \end{bmatrix} \quad (70)$$

where  $\tilde{\mathbf{T}}(k) = \mathbf{T}(k+1) \mathbf{T}^{-1}(k)$ . Design the following sliding surface,

$$s(k) = \bar{\mathbf{C}}(k) \phi(k) \quad (71)$$

where  $\bar{\mathbf{C}}(k) \in \mathbb{R}^{2 \times 4}$ ,  $\bar{\mathbf{C}}(k)$  can be divided into two sub-matrices



$$\bar{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{I}_2 \end{bmatrix} \quad (72)$$

where  $\mathbf{I}_2$  is a reversible constant matrix and  $\mathbf{C}_1$  is

$$\mathbf{C}_1 = \frac{\mathbf{I}_2 - \tilde{\mathbf{T}}(k)^{-1} \mathbf{C}_{1p}}{\sigma} \quad (73)$$

and  $\mathbf{C}_{1p} = \text{diag}(c_{11}, c_{22})$  and  $c_{11} \in (0,1)$ ,  $c_{22} \in (0,1)$ . We first demonstrate that the IQSM is stable.

From Eqns (67), (69) and (70) we have

$$\phi_h(k+1) = \tilde{\mathbf{T}}(k)\phi_h(k) + \sigma \tilde{\mathbf{T}}(k)\phi_l(k) \quad (74)$$

When  $s(k) = 0$ , the equation of the IQSM is

$$\phi_l(k) + \frac{(\mathbf{I}_2 - \tilde{\mathbf{T}}(k)^{-1} \mathbf{C}_{1p})}{\sigma} \phi_h(k) = 0 \quad (75)$$

Substituting Eqn (75) into (74) we have

$$\phi_h(k+1) = \mathbf{C}_{1p}\phi_h(k) \quad (76)$$

, showing that the IQSM is stable. The job to be done is to design a control law driving the system into the QSMB. For MIMO system the reaching law (47) is modified to

$$\frac{s(k+1) - s(k)}{\sigma} = -\bar{\eta} \text{sign}(s(k)) - \bar{q}s(k) \quad (77)$$

where  $\bar{\eta} = \begin{bmatrix} \eta^t & 0 \\ 0 & \eta^n \end{bmatrix}$  and  $\bar{q} = \begin{bmatrix} q^t & 0 \\ 0 & q^n \end{bmatrix}$  are two gain matrices. Manipulating reaching law (77)

yields

$$s(k+1) = -\sigma\bar{\eta}\text{sign}(s(k)) + \bar{Q}s(k) \quad (78)$$

where  $\bar{Q} = \begin{bmatrix} 1 - \sigma q^t & 0 \\ 0 & 1 - \sigma q^n \end{bmatrix}$ . Substituting Eqn (69) and (71) into Eqn (78)

$$\bar{C}\bar{A}_w\phi(k) - \bar{C}\bar{T}(k+1)\bar{B}_w u_{FB}(k) = -\bar{\eta}\sigma\text{sign}(s(k)) + (\mathbf{I}_2 - \bar{q}\sigma)\bar{C}\phi(k) \quad (79)$$

Solving Eqn (79) yields the following expression of the feedback controller

$$u_{FB}(k) = (\bar{C}\bar{T}(k+1)\bar{B}_w)^{-1}((\bar{C}\bar{A}_w - \bar{Q}\bar{C})\phi(k) + \eta\sigma\text{sign}(s(k))) \quad (80)$$

### 3.3 DSMCC for Robust Performance

So far the DSMCC has been designed for the nominal system. Next we consider the disturbance in the controller design. The nominal plant model (65) is modified to include the disturbances.

$$\phi(k+1) = \bar{A}_w(k)\phi(k) - \bar{T}(k+1)\bar{B}_w u_{FB}(k) + d_w(k) \quad (81)$$

where  $d_w(k) = F^{-1}\bar{T}(k+1)d(k)$  is the lumped disturbance. Substituting Eqn (80) into the disturbed system we can obtain the reaching law for the disturbed system

$$s(k+1) = -\sigma\bar{\eta}\text{sign}(s(k)) + \bar{Q}s(k) - \bar{c}d_w(k). \quad (82)$$

It is clear that the disturbance changes the dynamics of the sliding mode. Though the disturbance is in general unmolded, it is reasonable to assume that it is bounded.

$$D_l < \bar{c}d_w(k) < D_u$$

where  $D_l$  and  $D_u$  are lower and upper boundaries of the lumped disturbances. An implementable robust DSMCC is given by

$$u_{FB}(k) = -(\bar{C}\bar{T}(k+1)\bar{B})^{-1}((\bar{C}\bar{A}_w(k) - (I_2 - q\sigma)\bar{C})\phi(k) + \eta\sigma\text{sign}(s(k))) - (\bar{C}\bar{T}(k+1)\bar{B})^{-1}\left(\frac{D_u + D_l}{2} + \frac{\text{sign}(s(k))(D_u - D_l)}{2}\right) \quad (83)$$

It should be noted the properties of the idea quasi sliding mode will be lost due to the disturbances [83]. Control law (83) is capable of driving the sliding surface into the QSMB and thus stabilizing the system. However, the width of the QSMB is wider than the undisturbed case and as a result the steady state error is increased. Therefore switching gain should be increased cautiously.

### 3.4 Experimental Validations

The *X-axis* and *Y-axis* of the micro mill are used in this experiment. Both axes are driven by DPRANIE-015A and the gains for *X-axis* and *Y-axis* are

$$b_x = 200.2\text{mm} / (\text{Vs}^2), \quad b_y = 341.7\text{mm} / (\text{Vs}^2)$$

The following control algorithms are compared:

Proportional-Integral-Differential (PID) control. The controller gains for the *X-axis* are  $k_p = 600, k_d = 2.5$  and  $k_i = 50$ ; for the *Y-axis*,  $k_p = 400, k_p = 1.6$  and  $k_i = 40$ . The control parameters are chosen to achieve almost matched dynamics in the two axis with pairs of dominant poles of nature frequencies around  $346\text{rad/s}$  and  $369\text{rad/s}$  and damping ratios around 0.72 and 0.74 for the *X-axis* and *Y-axis*, respectively.

DSMCC. The parameters are chosen as  $C_{1p} = \text{diag}[0.8, 0.6]$ ,  $q = \text{diag}[200, 400]$ , switching gain  $\eta = \text{diag}[0, 3]$ . Notice that the normal direction is assigned with faster dynamics than the tangent direction. A first order low pass filter is used to alleviate the chatters caused by the switching.

### 3.4.1 Performance Index

The following indexes are used to evaluate the performance of the controllers.

1.  $|\varepsilon_{max}|$  - Maximum of the absolute value of the contour errors.
2.  $|\varepsilon_{mean}|$  - The absolute value of the mean of the contour errors is used to measure the average contour following performance.
3.  $\varepsilon_{std}$  - Standard deviations of the contour errors are used to evaluate the smoothness of the motions.

### 3.4.2 Experiment 1 Guitar Contour

The “Cutaway guitar” contour is designed, see Figure 32. The lead-in and lead-out parts are not included. Both PID controller and DSMCC are tested. The actual contour errors are calculated offline.

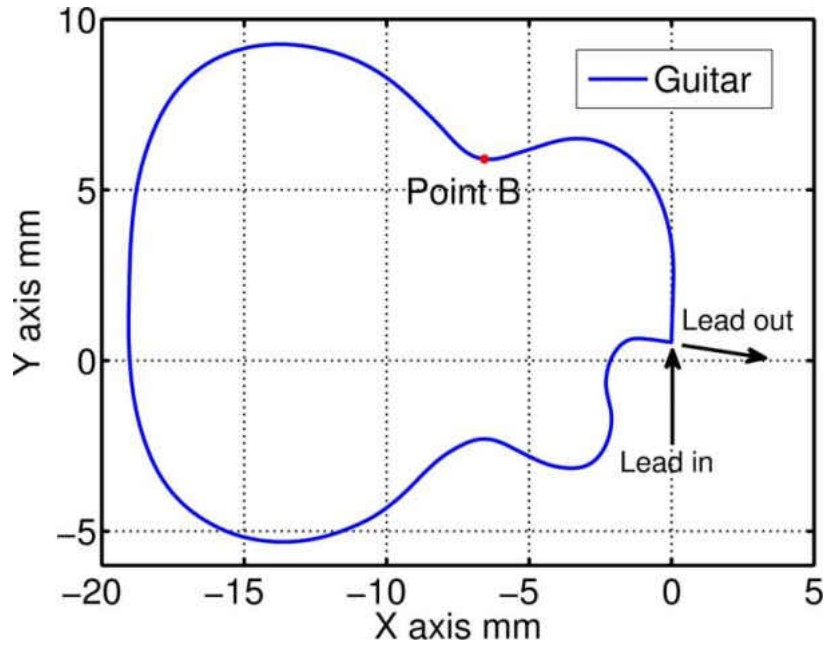


Figure 32: Cutaway guitar contour

Case A: Guitar contour at a feedrate of  $20\text{mm/s}$ . The results are shown in Figure 33. DSMCC clearly achieves much better performances than PID controller. The maximum contour error of the PID controller is  $7.64\mu\text{m}$  while for DSMCC, the maximum contour error is only  $1.60\mu\text{m}$ .

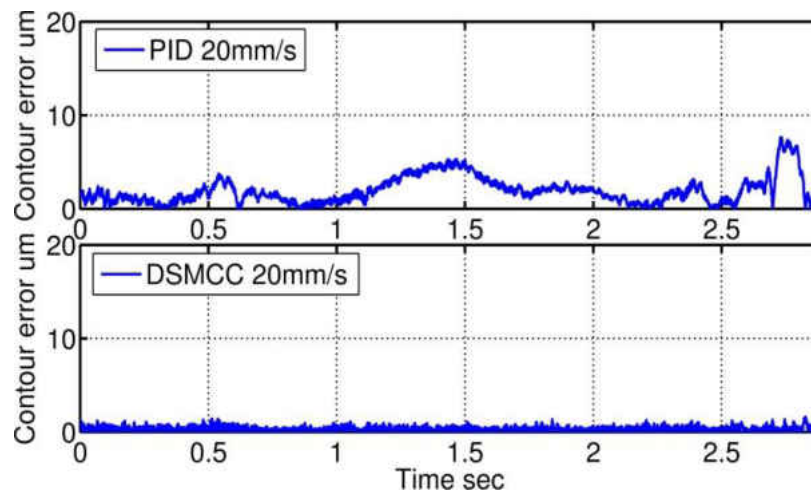


Figure 33: Contour error of guitar contour at  $20\text{mm/s}$

*Case B:* For comparison purpose, guitar contour tracking is repeated at a higher feedrate of  $30\text{mm/s}$ . The results are shown in Figure 34. The statistics of the guitar contour following experiments are listed in Table 5. Compared to the same experiment conducted at  $20\text{mm/s}$ , the maximum error of PID controller dramatically increases to  $14.93\mu\text{m}$  and the absolute mean contour error increase to  $2.58\mu\text{m}$ . Again DSMCC beat PID controller by considerable margin with a maximum error of  $3.44\mu\text{m}$  and a mean error of  $0.44\mu\text{m}$ . By comparing the performance of DSMCC in both cases, it is clear that by introducing the discrete task coordinate, DSMCC achieves contour following performance almost independent to the contours' shape and changes in feedrate.

Table 5: Guitar contour results

<i>Guitar</i>		$ \varepsilon_{max} $	$\varepsilon_{std}$	$ \varepsilon_{mean} $
<i>Case A</i>	PID	$7.64\mu\text{m}$	$1.44\mu\text{m}$	$1.95\mu\text{m}$
	DSMCC	$1.60\mu\text{m}$	$0.25\mu\text{m}$	$0.33\mu\text{m}$
<i>Case B</i>	PID	$14.93\mu\text{m}$	$2.70\mu\text{m}$	$2.58\mu\text{m}$
	DSMCC	$3.44\mu\text{m}$	$0.41\mu\text{m}$	$0.44\mu\text{m}$

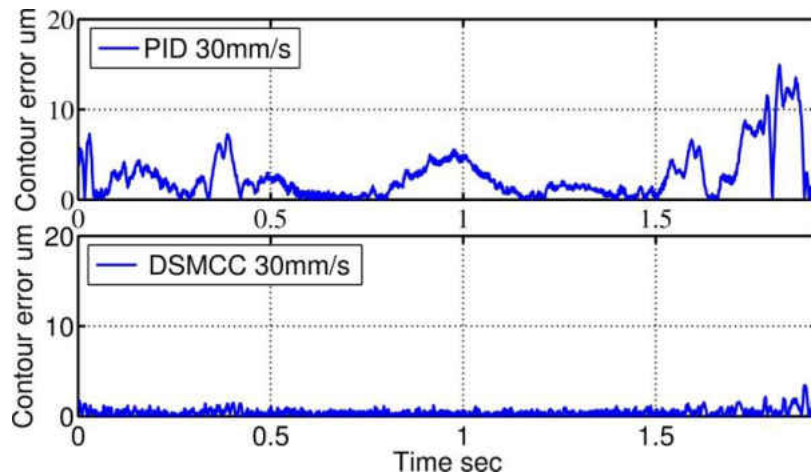


Figure 34: Contour error of guitar contour at  $30\text{mm/s}$

From the guitar contour experiment, it is safe to draw the following conclusions:

1. DSMCC performs better than PID controller in terms of all performance indexes.
2. Changes in the contour shapes and feedrate lead to significant performance variations in PID controller due to the lack of coordination between axes. On the other hand, DSMCC shows good consistency in all experiments. This merit makes DSMCC especially suitable for complex contour following applications.

### 3.4.3 Experiment 2 Dog Bone Contour

The ‘dog bone’ contour is designed, see Figure 35.

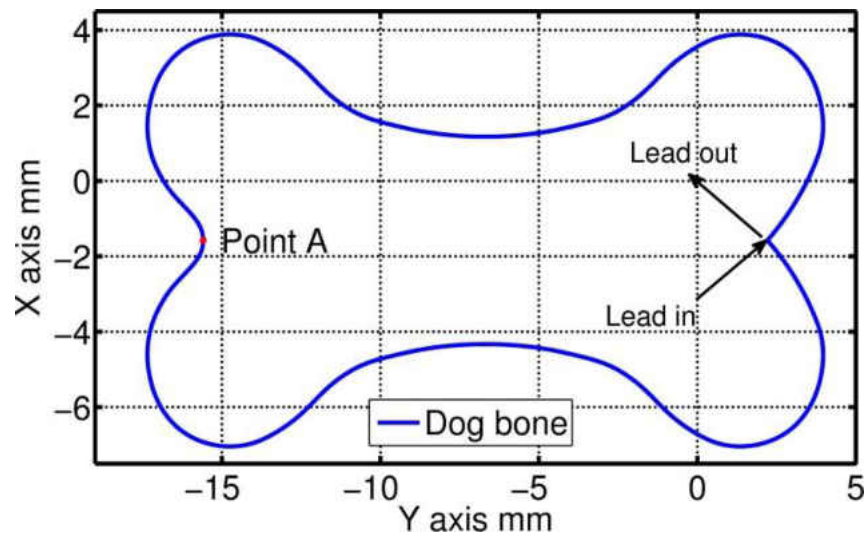


Figure 35: Dog bone contour

*Case A:* Dog bone contour at a feedrate of  $20\text{mm/s}$ . The contour error comparisons are illustrated in Figure 36. The proposed DSMCC exhibits a superior contour following performance to the PID controller.

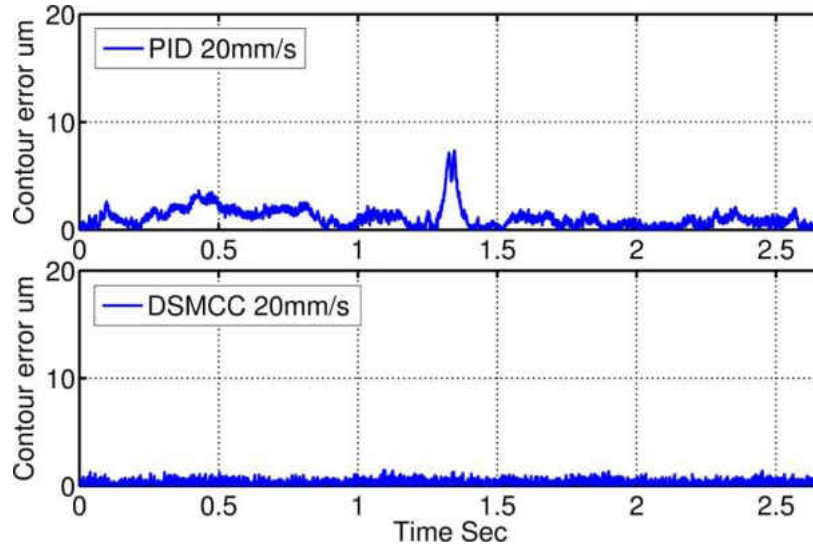


Figure 36: Contour error of dog bone contour at  $20\text{mm/s}$

*Case B:* To test the influences of the feedrate on the contour performances, the dog bone contour following experiment is repeated at a higher feedrate of  $30\text{mm/s}$ . The results are illustrated in Figure 37. The contour error of the PID controller increases considerably as the feedrate increases, while the DSMCC shows very little performance change. In addition, PID contour error shows a symmetric patterns in Figure 36 and Figure 37, which is resulted from the symmetry of the dog bone contour. This clearly shows that the both the shapes of the contours and feedrate influence the performances of the decoupled controller. On the other hand, those factors show little effects on the proposed DSMCC. The results of the dog bone contour following experiment are listed in Table 6. DSMCC performs better than PID controller in terms of all performance indexes.



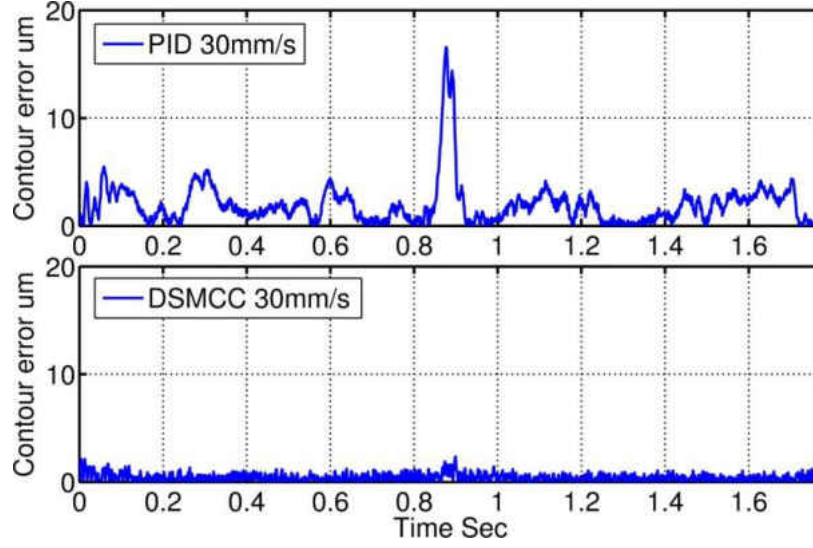


Figure 37: Contour error of Dog bone contour at 30mm/s

Table 6: Dog bone contour results

<i>Dog Bone</i>		$ \epsilon_{max} $	$\epsilon_{std}$	$ \epsilon_{mean} $
<i>Case A</i>	PID	$7.35 \mu m$	$0.95 \mu m$	$1.10 \mu m$
	DSMCC	$1.46 \mu m$	$0.26 \mu m$	$0.36 \mu m$
<i>Case B</i>	PID	$16.66 \mu m$	$2.00 \mu m$	$1.94 \mu m$
	DSMCC	$2.35 \mu m$	$0.40 \mu m$	$0.33 \mu m$

#### 3.4.4 Experiment 3 Performance with Perturbations and Uncertainties

Next we would like to test the robust performance of both controllers. Circular contour is chosen for this part of experiment. The advantages of using circular contours to test the contour following performances are elaborated in [85]. Circular contour is also preferred for its simplicity to calculate contour error

$$\epsilon_c^n = R_c - ((x - x_c)^2 + (y - y_c)^2)^{0.5} \quad (84)$$

where  $R_c$  is the radius;  $x_c$  and  $y_c$  define the center of the circular contour. The simplicity of Eqn (84) facilitates the performance analysis. The X-Y table is commanded to follow the circular contour described by

$$\begin{bmatrix} x(t) = 2 \sin(6\pi t) \\ y(t) = 2 \sin(6\pi t) - 2 \end{bmatrix} \quad (85)$$

The desired feedrate is  $37.69\text{mm/s}$ . The parameters of the circular contour are carefully chosen so the winding currents in the  $X$ -axis ( $X$ -axis is the lower axis of the table and has to overcome more inertia during motion than the  $Y$ -axis) does not exceed the rated continuous current. The following experiments are performed:

*Case A:* To test the nominal contour following performance, the experiment is conducted with no disturbances or additional payloads. The results are illustrated in Figure 38. The maximum contour error ( $7.95 \mu\text{m}$ ) of the PID controller is significantly larger than that of the contour DSMCC ( $1.11\mu\text{m}$ ).

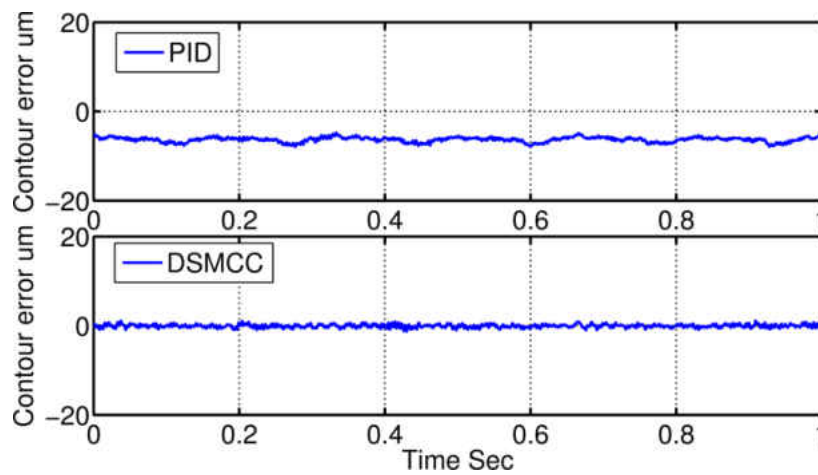


Figure 38: Circular contour following under nominal condition

*Case B:* The experiment was repeated with an additional 4 kg payload mounted on the Y-axis. Figure 39 shows the results of the experiment.

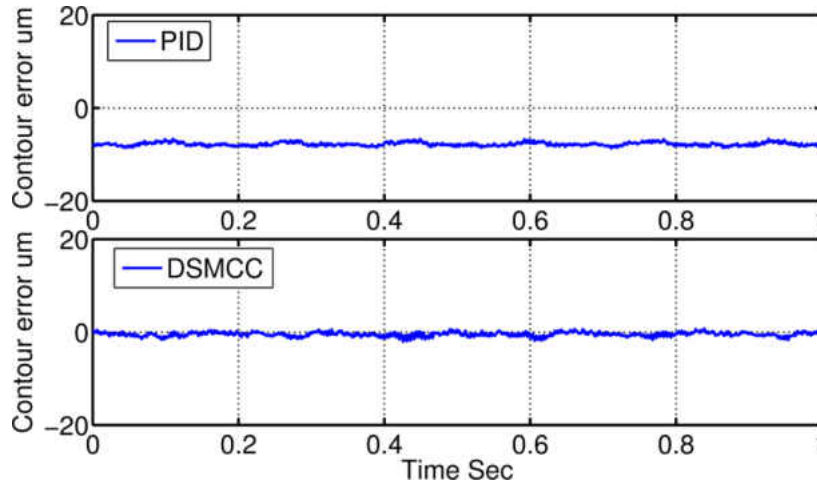


Figure 39: Circular contour following with a 4Kg payload

For the PID controller, the payload causes the maximum value of the contour error to increase to  $8.7\mu m$  and the mean contour error to increase to  $7.78\mu m$ . On the other hand, the performance the DSMCC shows slight worse performance than the undisturbed case and beats the PID controller by considerable margin.

*Case C:* A large step disturbance (a simulated disturbance equivalent to a winding current of 1.05A, which is about 50% of the motor's rated continuous current) is directly injected to the current loop of the X-axis at  $t_1=0.083s$  and then removed at  $t_2=0.75s$ .  $t_1$  and  $t_2$  are chosen such that the sudden changes of disturbance directly acts on the normal directions of the circle. The experimental results are shown in Figure 40.

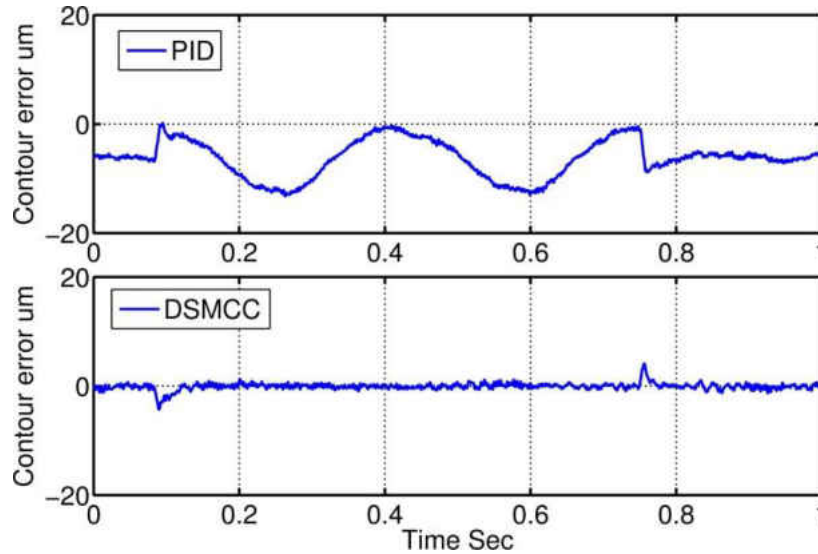


Figure 40: Circular contour following with large step disturbance

The disadvantage of the PID controller can be clearly observed in Figure 40 at the time when the disturbance is applied. Due to the lack of coordination, the standard variation of contour error significantly increases to  $3.38\mu m$ , compared to  $0.57\mu m$  in the nominal case. On the other hand, the disturbance does not affect the performance of DSMCC except for the transients when the sudden changes of the disturbances happen. It should be noted that the transients are inevitable since the step disturbance has frequency components beyond the Nyquist frequency of the control system.

### 3.5 Summary

In this chapter, a novel discrete sliding contour control has been developed. The proposed controller extends the traditional DSMC, which has been limited for single axis applications, to bi-axial contour following problems. Compared to the traditional controller, the proposed controller achieves consistent contour following performance regardless of the changes in

contour shapes and feedrate. In addition, the proposed controller inherits robustness from the DSMC and is immune to disturbances and uncertainties. In the next chapter we will extend the bi-axial controller for contours in  $\mathbb{R}^3$  following problems and test it in real machining applications.

## CHAPTER 4      MOVING FRAME OPTIMIZATION OF REGULAR CURVES IN $\mathbb{R}^3$

### 4.1    Background Knowledge

Theoretically, the shortest distance between a curve and the actual cutter location can be exactly calculated. However, it is difficult to implement such calculation in realtime especially for complex curves. Therefore the contour errors are typically obtained by estimations. As previously introduced, there are in general two approaches of estimating contour error. The first estimation is performed in the world frame [36, 82, 86] while the other one is established with respect to a moving frame attached to the desired contour. The global frames are mostly chosen to be stationary polar or curvilinear coordinates so the controller design can be simplified. Nevertheless the global frame cannot guarantee orthogonality of the coordinates. As such, this method has been mostly tested used on bi-axial circular or elliptical contours [36, 87]. On the contrary, the moving frame is established based on the local properties of the desired contour. Since the contour error is typically based on the local property of the desired contours, moving frame can achieve much more accurate estimation than the global frame method. Therefore the local frame scheme is preferred in this study.

Local frame estimations have found applications in both feedforward and feedback type of contour control. It should be noted that the local frame plays different roles in the two contour control schemes previously discussed. For the feedforward contour control, the purpose of the local frame is solely for the contour error estimation while for the feedback type contour control the dynamics of the local frame is trajectory dependent and appears in the resulted control system. Therefore, in case of the feedback type contour control, the local frame should be of

smooth dynamics to avoid spiky control efforts. For the research focusing on planar contours (bi-axial contours or contours fixed on inclined planes), the choice of the local frame is unique and intuitive. However, it is not always true for regular contours in  $\mathbb{R}^3$ . We will show that curve's torsion should be considered when designing the moving frame for  $\mathbb{R}^3$  curves and an optimization method to smoothen the transitions of the local frames will be presented. In addition, the local frame method suffers a significant contour error due to the linear approximation (tangent vector approximation). For planar contours, the contour error can be better estimated by constructing a substitute curve using both direction and instantaneous curvature information of the actual curve (circular approximation). By the same token, a precision approximation of  $\mathbb{R}^3$  contours should incorporate direction, curvature and torsion. However, it is not always desirable to do so because 1) a substitute curve satisfying the requirements may be difficult to obtain 2) the shortest distance between the substitute curve and the actual position may be difficult to calculate. To avoid the complicated estimation procedure, we propose a new substitute contour error that increases the contour following performance. The proposed method only needs the information of the local frame and command feedrate.

Based on the work previously presented, a two loop contour controller has been designed as well. The out loop is based on the optimized local frame to decouple the dynamics for tangential component and normal components of the error vector. The inner loop is designed based on integral sliding mode (ISMC) to achieve chatter free robustness. The effectiveness of the proposed method is examined by comparative experiments using our *3-axis* micro mill.

*Notations:* Denote  $\mathbf{G}$  as the fixed machine coordinate,  $\mathbf{F}'$  and  $\mathbf{F}$  is the non-optimized and optimized local frame, respectively. A vector has different representations with respect to these

three different frames. Thus the following convention is established to differentiate the three representations

1. Non-primed capital case denotes the vector's representation with respect to  $\mathbf{G}$
2. primed low case denotes the vector's representation with respect to  $\mathbf{F}'$
3. non-primed low case stands for the vector's representation with respect to  $\mathbf{F}$ .

## 4.2 Problem Formulation

In this section, we first introduce the contour following problem in  $\mathbb{R}^3$  and the construction of local coordinate, before presenting the new optimization method.

### 4.2.1 Local Frame of Regular Contours in $\mathbb{R}^3$

The desired contour in  $\mathbb{R}^3$  is given by  $R(t) = [R_X(t) R_Y(t) R_Z(t)]^T$ . For any commanded position  $D$  on  $R$ , the local frame  $\mathbf{F}'$  is defined using the moving trihedron [46, 47]

$$\mathbf{F}'(t) := \{t'(t) \quad n'(t) \quad b'(t)\} \quad (86)$$

where  $t'(t)$  is the tangent vector,  $n'(t)$  stands for the principal normal vector and  $b'(t)$  is the bi-normal vector. The plane spanned by the  $n'(t)$  and  $t'(t)$  is referred to as the osculating plane. The schematic diagram of the local frame is illustrated in Figure 41. The tracking error vector  $\mathbf{E}(t)$  is

$$\mathbf{E}(t) = D(t) - Q(t) \quad (87)$$

where  $Q(t) = [q_X(t) q_Y(t) q_Z(t)]^T$  stands for the actual position vector. For simplicity,  $t$  is omitted thereafter. The representation of the error vector with respect to  $\mathbf{F}'$  is given by



$$\mathbf{e}' := [e'_t \quad e'_n \quad e'_b]^T = \mathbf{T}'\mathbf{E} \quad (88)$$

where

$$\mathbf{T}' = (\mathbf{F}')^{-1} \quad (89)$$

is the transformation matrix between the vector representations with respect to different frames.

$\mathbf{T}' \in \mathbb{R}^{3 \times 3}$  is unitary by definition.

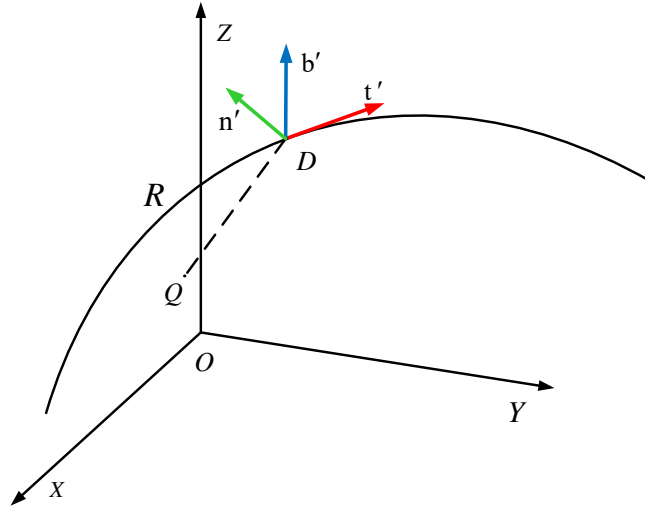


Figure 41: Local frame of three dimensional contours

By construction,  $e'_b$  and  $e'_n$  are normal error components or the contour error.  $e'_t$  is the tangent error component. For conciseness, the following assumptions are adopted in this chapter.

*Assumption 1:*

1.  $R$  is parameterized by time and is at least  $C^2$  continuous.
2.  $\dot{\mathbf{E}}$  and  $\ddot{\mathbf{E}}$  are bounded. Taking the first derivatives of Eqn (88) yields

$$\dot{\mathbf{e}}' = (\dot{\mathbf{F}}')^T \mathbf{E} + (\mathbf{F}')^T \dot{\mathbf{E}} \quad (90)$$

which is the first order error dynamics [47, 87, 88]. From Eqn (90), it is clear that the dynamics of contour error depends on the not only the individual error dynamics but also the dynamics of the moving trihedron (local frame).

3. The tangential error  $e_t' \ll \rho$ ,  $\rho$  is the instantaneous radius to of R.

#### 4.2.2 Optimization of Local Frame

To the author's best knowledge, local frame (86) has been successfully applied to two cases. The first case is the well-studied bi-axial contour following problem while the other studies contour on certain fixed planes in  $\mathbb{R}^3$ . These two cases are essentially equivalent because they both study planar motions and the task frame (86) defined using the moving trihedron is not problematic. However,  $\mathbf{F}'$  is not always applicable for regular contours in  $\mathbb{R}^3$ . One difference between the planar contours and regular  $\mathbb{R}^3$  contours is torsion, which describes how fast a curve twists out of the plane of curvature. Torsion  $\tau$  is defined as [89]

$$\dot{\mathbf{b}}' = -\tau \mathbf{n}' \quad (91)$$

$\dot{\mathbf{b}}'$  indicates how fast  $\mathbf{F}'$  rotates about the instantaneous tangent vector and clearly  $\mathbf{F}'$  rotates sharply when  $\tau$  is large. From Eqn (90), the dynamics of  $\mathbf{F}'$  is reflected in the resulted error dynamics and an overly fast-rotating  $\mathbf{F}'$  may cause spiky control or even saturations.  $\mathbf{F}'$  works flawlessly on planar contours because  $\tau$  is constantly zero for planar contour. However, in  $\mathbb{R}^3$  contours, torsion can be large even the contours are smooth.

Unlike the planar contours, the choice of the local frame for regular  $\mathbb{R}^3$  contours is not unique. This allows some freedom for optimizations. In practice the desired contours are

sampled at a certain sampling frequency and we next present the optimization method that leads to a series of local frames with smooth transition for the sampled contour  $R(kt_s) = [R_x(kt_s) R_y(kt_s) R_z(kt_s)]^T$ , where  $k \in \mathbb{Z}^+$  is the time index and  $t_s$  is the sampling time. For simplicity, sub index  $k$  is used to replace  $kt_s$ , for instance  $R_k = R(kt_s)$ . For any commanded position  $R_k$ , the non-optimized task frame obtained by (86) is denoted as  $\mathbf{F}'_k$  and the corresponding optimized task frame is  $\mathbf{F}_k$ .

$$\mathbf{F}'_k := \{t_k \quad n_k \quad b_k\} \quad (92)$$

The numerical approximation of  $\dot{\mathbf{F}}'$  is given by

$$\dot{\mathbf{F}}' = \frac{\mathbf{F}'(k+1) - \mathbf{F}'(k)}{\sigma} = \frac{\Delta \mathbf{F}'(k)}{\sigma} \quad (93)$$

and

$$(\dot{\mathbf{F}}')^T \mathbf{E} = \begin{bmatrix} (t'_{k+1} - t'_k)^T \bullet \mathbf{E} / \sigma \\ (n'_{k+1} - n'_k)^T \bullet \mathbf{E} / \sigma \\ (b'_{k+1} - b'_k)^T \bullet \mathbf{E} / \sigma \end{bmatrix} \begin{cases} \text{Tangent direction} \\ \text{Normal direction 1} \\ \text{Normal direction 2} \end{cases} \quad (94)$$

It should be noted that though there are infinite choices for the local frame, the tangent vector is unique and solely determined by the velocity vector at  $R_k$ . As such, the tangent vector cannot be altered. Recall the following inequalities

$$\begin{aligned} (n'_{k+1} - n'_k)^T \bullet \mathbf{E} &\leq |(n'_{k+1} - n'_k)| |\mathbf{E}| = |\Delta n_k| |\mathbf{E}| \\ (b'_{k+1} - b'_k)^T \bullet \mathbf{E} &\leq |(b'_{k+1} - b'_k)| |\mathbf{E}| = |\Delta b_k| |\mathbf{E}| \end{aligned} \quad (95)$$

From inequalities (95), one can see that to avoid drastic dynamic changes in the normal direction caused by the unnecessary rotation of the moving frame, the goal of the optimization is to obtain the minimum of the following cost function

$$g = |n_{k+1} - n_k|^2 + |b_{k+1} - b_k|^2 \quad (96)$$

Recall that the tangent vector at any given reference position  $R_k$  is fixed and therefore  $\mathbf{F}'_k$  and  $\mathbf{F}_k$  in fact share the same tangent vector, which means  $\mathbf{F}_k$  can be obtained by rotating  $\mathbf{F}'_k$  and the optimization problem becomes finding the optimized rotation angle  $\theta_k$ . We begin the optimization procedure with choosing  $\mathbf{F}_1 = \mathbf{F}'_1$  and rotating  $\mathbf{F}'_2$  to obtain the optimized frame  $\mathbf{F}_2$

$$\mathbf{F}_2 = \{t_2 \quad n_2 \quad b_2\} = \mathbf{F}'_2 \mathbf{T}_2 \quad (97)$$

where  $\mathbf{T}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_2 & \sin\theta_2 \\ 0 & -\sin\theta_2 & \cos\theta_2 \end{bmatrix}$ .

Note that the tangent vector  $t'_2$  remains unchanged in between the transformation, i.e.,  $t'_2 = t_2$ . The schematic of the transformation is illustrated in Figure 42.

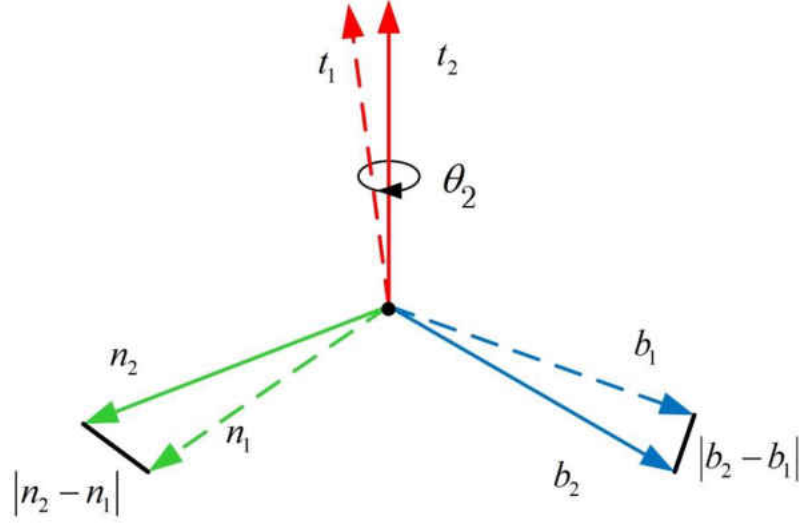


Figure 42: Schematic diagram of the local frame rotation

To yield the smoothest transition between the two consecutive frames, the rotation angle  $\theta_2$  is optimized by minimizing the following quadratic cost function

$$g(\theta_2) = |n_2(\theta_2) - n_1|^2 + |b_2(\theta_2) - b_1|^2 \quad (98)$$

After some mathematic manipulations,  $g(\theta_2)$  can be simplified to

$$g(\theta_2) = 4 + 2k_1 \sin(\theta_2) - 2k_2 \cos(\theta_2) \quad (99)$$

where  $k_1 = b_1 \cdot n_2' - b_2' \cdot n_1$  and  $k_2 = b_1 \cdot b_2' + n_2' \cdot n_1$ . Setting the first derivative of  $g(\theta_2)$  to zero

$$\frac{dg(\theta_2)}{d\theta_2} = 2k_1 \cos \theta_2 + 2k_2 \sin \theta_2 = 0 \quad (100)$$

we have

$$\theta_2 = \tan^{-1}\left(-\frac{k_1}{k_2}\right) \quad (101)$$

The solutions to Eqn (101) must also ensure  $\frac{dg^2(\theta_2)}{d\theta_2^2} > 0$  for the minimum of  $g(\theta_2)$ .  $\mathbf{F}_2$  can be calculated by Eqn (97) using the optimized rotation angle. Once  $\mathbf{F}_2$  is determined, the optimization can be repeated for the rest local frames. The dynamics of the error vector with respect to frame  $\mathbf{F}$ , denoted as  $\mathbf{e} = [e_t \ e_n \ e_b]^T$  is given by

$$\mathbf{T}\ddot{\mathbf{E}} = \mathbf{T}\ddot{\mathbf{T}}^{-1}\mathbf{e} + 2\mathbf{T}\dot{\mathbf{T}}^{-1}\dot{\mathbf{e}} + \ddot{\mathbf{e}} \quad (102)$$

where  $\mathbf{T} = \mathbf{F}^{-1}$ .

#### 4.3 Contour Error Estimation of $\mathbb{R}^3$ Contours in Local Frame

Theoretically, the contour error can be exactly calculated for arbitrary curves. Yet it is difficult to implement in realtime unless the curve is simple. Thus the desired contours are normally approximated using simple curves that have analytical expression of the contour error. The curves in  $\mathbb{R}^3$  can be in general characterized by direction (tangent vector), curvature (rate of change of tangent vector) and torsion (rate of change of the principal normal vector). The task frame based method is a linear approximation that only uses the direction information of the desired curve, i.e., the tangent vector. Therefore it is poor for the linear approach to approximate curves with nonzero curvature [37]. By the same token, the circular approach that includes the direction and curvature information is only accurate for zero-torsion curves (plane curves).

From the previous discussions, a precision approximation of a regular curve in  $\mathbb{R}^3$  should consider direction, curvature and torsion. However, this might be difficult because 1) a simple curve that satisfies all the requirements may not exist 2) even if such a simple curve is achievable, the shortest distance between the simple curve and the actual position may still be difficult to solve, which defeats the purpose of approximation. To circumvent this problem, we present a simple estimation method for regular curves in  $\mathbb{R}^3$  without computing too many geometric parameters. The proposed method is established with respect to the optimized task frame previously developed, see Figure 43. Plane QNB is normal to tangent vector  $t$  and passes the actual position  $Q$ . Both  $t$  and  $R$  intersect with QNB and the intersections are denoted as  $P$  and  $N$ , respectively. By comparing the construction of plane QNB to Eqn (102), we have

$$e_t = |\overrightarrow{ND}|, \quad e_b = \overrightarrow{QN} \bullet b, \quad e_n = \overrightarrow{QN} \bullet n \quad (103)$$

where  $\bullet$  denotes the inner product of vectors. From Figure 43,  $\overrightarrow{QN}$  is the normal error obtained from the local frame.

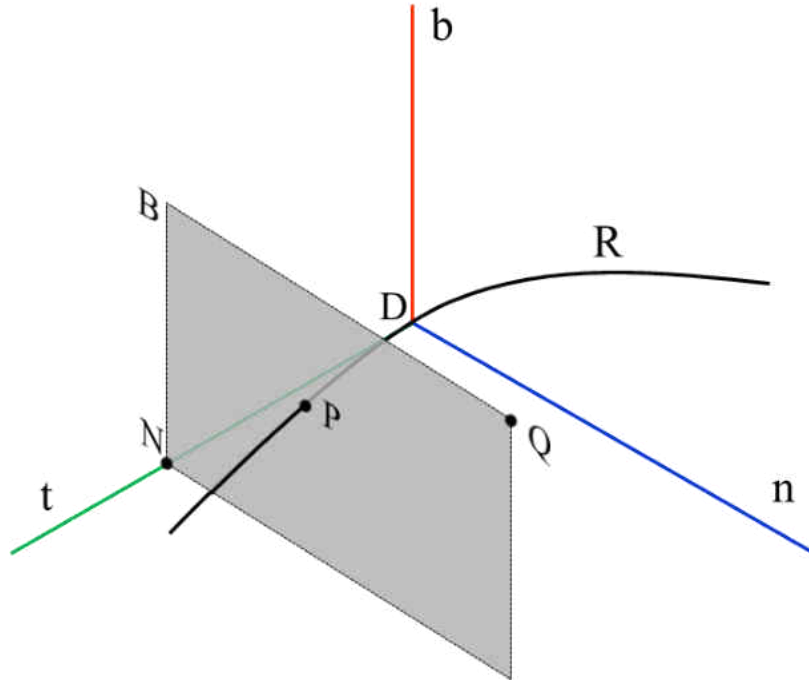


Figure 43: Schematic diagram of equivalent contour error in  $\mathbb{R}^3$

Since the control efforts along the 3 basis vectors are decoupled, the normal component of the control effort is always within the plane QNB and forces Q toward N. N seldomly coincides with P and the weakness of the linear approximation is very clear. One can observe that, within the normal plane QNB, the actual position approaches the desired contour only if Q approaches P and the elimination of the contour error is equivalent to  $|\overline{PQ}| = 0$ . From this perspective,  $\overline{PQ}$  can be deemed as an alternative measurement of the exact contour error. Therefore, instead of directly estimating the actual contour error, which can be difficult for curves in  $\mathbb{R}^3$ ,  $\overline{PQ}$  is used as a substitute of the actual contour error. One needs to know the coordinate of point P to determine  $\overline{PQ}$ . Denote R with respect to the local frame  $\mathbf{F}$  as r

$$\mathbf{r}(t) := [r_t(t) \ r_n(t) \ r_b(t)]^T = \mathbf{T}(\mathbf{R}(t) - \mathbf{D}) \quad (104)$$



From the first equation of Eqn (103), the time parameter  $t$  that corresponds to P must satisfy

$$r_t(t) = e_t \quad (105)$$

Eqn (105) may be difficult to solve unless the contour is simple. Thus  $r_t(t)$  is approximated by its first order Taylor's expansion at D

$$r_t(t) = \dot{r}_t(t_D) \cdot (t - t_D) \quad (106)$$

where  $t_D$  denotes the current time parameter corresponding to D. We are only interested in the geometric relationship and therefore  $\mathbf{F}$  is considered as a stationary frame, i.e.,

$$\dot{r}(t) = \mathbf{TR}(t) \quad (107)$$

Here the first order Taylor's expansion (106) is used for in favor of its simplicity. The estimation of  $t$ , denoted as  $\hat{t}$ , can be solved from Eqn (105), (106) and (107) if  $e_t$  is known. The estimation of coordinate of point P is solved by substituting  $\hat{t}$  into Eqn (104) and the estimation of  $\overline{PQ}$ , denoted as  $\overline{PQ}$ , can be obtained. The accuracy of  $\overline{PQ}$  can be improved by introducing quadratic term into (106) (R is twice order differentiable from assumption 1). Similar to Eqn (103), we can define

$$\varepsilon := \begin{bmatrix} \varepsilon_t & \varepsilon_n & \varepsilon_b \end{bmatrix}^T = \begin{bmatrix} e_t & \overline{PQ} \cdot n & \overline{PQ} \cdot b \end{bmatrix}^T \quad (108)$$

$\varepsilon$  is the vector of the substitute contour error.  $\varepsilon$  and  $e$  are both defined with respect to  $\mathbf{F}$  and thus  $\varepsilon$  is better than  $e$  because it reduces the overcut or undercut. Denote the corrections brought by  $\varepsilon$  as

$$\eta := [0 \ \eta_n \ \eta_b]^T = \varepsilon - e \quad (109)$$

Substituting Eqn (109) into Eqn (102), we have

$$\mathbf{T}\ddot{\mathbf{E}} = (\ddot{\varepsilon} - \ddot{\eta}) + 2\mathbf{T}\dot{\mathbf{T}}^{-1}(\dot{\varepsilon} - \dot{\eta}) + \mathbf{T}\ddot{\mathbf{T}}^{-1}(\varepsilon - \eta) \quad (110)$$

Eqn (110) is the error dynamics represented by the substitute error dynamics.

#### 4.4 Controller Design

The optimized task frame enables the controller designer to decouple the tangent and normal error dynamics and focus more on the normal direction. Looking at the right hand side of Eqn (110) we introduce a virtual control  $U_c$  denoting the desired control effort with respect to the task frame. The purpose of introducing  $U_c$  is to achieve the following time invariant dynamics of  $\varepsilon$

$$\ddot{\varepsilon} + \mathbf{K}_D \dot{\varepsilon} + \mathbf{K}_P \varepsilon = 0 \quad (111)$$

where  $\mathbf{K}_P = \text{diag}(k_{pt}, k_{pn}, k_{pb})$ ,  $\mathbf{K}_D = \text{diag}(k_{dt}, k_{dn}, k_{db})$  are positive definite diagonal matrices. Substituting Eqn (111) into the RHS of Eqn (110) and setting it to zero we can obtain the expression of the virtual control

$$U_c = -2\mathbf{T}\dot{\mathbf{T}}^{-1}\dot{\varepsilon} - \mathbf{T}\ddot{\mathbf{T}}^{-1}\varepsilon - \varphi + \mathbf{K}_d \dot{\varepsilon} + \mathbf{K}_p \varepsilon \quad (112)$$

where  $\varphi = \ddot{\eta} + 2\mathbf{T}\dot{\mathbf{T}}^{-1}\dot{\eta} + \mathbf{T}\ddot{\mathbf{T}}^{-1}\eta$  denotes the extra dynamics brought by the substitute contour error. Next we transfer the virtual controller back into the real system. Consider the dynamics of a 3-axis micro mill

$$\ddot{\mathbf{Q}} + \mathbf{C}\dot{\mathbf{Q}} = \mathbf{B}\mathbf{U} \quad (113)$$

where  $\mathbf{C} = \text{diag}(c_z, c_y, c_x)$ ,  $\mathbf{B} = \text{diag}(b_x, b_y, b_z)$ . Multiplying both side of Eqn of (113) by  $\mathbf{T}$  and noticing  $\mathbf{E} = \mathbf{R} - \mathbf{Q}$  we can obtain

$$\mathbf{T}\ddot{\mathbf{E}} + \mathbf{T}\mathbf{C}\dot{\mathbf{E}} + \mathbf{T}\mathbf{B}\mathbf{U} = \mathbf{T}\ddot{\mathbf{R}} + \mathbf{T}\mathbf{C}\dot{\mathbf{R}} \quad (114)$$

The proposed state feedback controller consist of two parts

$$\mathbf{U} = \mathbf{B}^{-1}(\mathbf{T}^{-1}\mathbf{U}_c + \mathbf{U}_f) \quad (115)$$

where  $\mathbf{B}^{-1}\mathbf{T}^{-1}\mathbf{U}_c$  is the virtual control transferred back to the machine coordinate. Substituting Eqn (115) into Eqn (114) we have

$$0 = \ddot{\boldsymbol{\varepsilon}} + \mathbf{K}_P\boldsymbol{\varepsilon} + \mathbf{K}_D\dot{\boldsymbol{\varepsilon}} + \mathbf{C}\dot{\mathbf{E}} - \ddot{\mathbf{R}} - \mathbf{C}\dot{\mathbf{R}} + \mathbf{B}\mathbf{U}_f \quad (116)$$

Solving Eqn (116) for  $\mathbf{U}_f$  that yields the desired error dynamics for  $\boldsymbol{\varepsilon}$  we can obtain

$$\mathbf{U}_f = -\mathbf{C}\dot{\mathbf{E}} + \ddot{\mathbf{R}} + \mathbf{C}\dot{\mathbf{R}} \quad (117)$$

#### 4.5 Robust Contour Control

Robust controllers has been proposed and implemented in the position loop [88]. However, this is not advisable for system since the system dynamics are time varying (LTV) w.r.t the task coordinate, making the parameters tuning a time-consuming task.

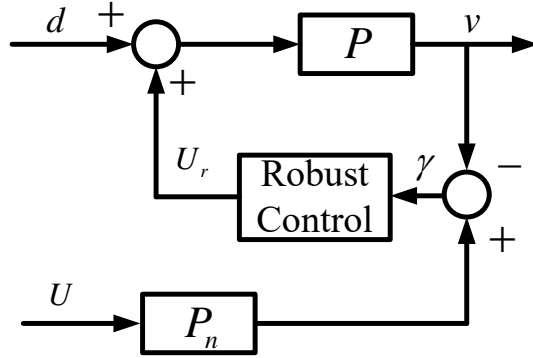


Figure 44: Block diagram of the robust control

An alternative is to augment the control system with a robust controller implemented in the velocity loop. The velocity loop is a disturbed linear time invariant (LTI) system w.r.t the machine coordinate and therefore the controller design is greatly simplified. The block diagram of the robust controller is shown in Figure 44, where  $u(s)$  is the control signal of the state feedback (115);  $\gamma$  stands for the error caused by lumped disturbances in the system;  $p_n(s)$  is the nominal transfer function of the velocity loop corresponding to (113)

$$p_n(s) = \frac{b}{s + c} \quad (118)$$

The objective of the robust controller is to force the actual velocity loop to behave like the nominal model, i.e. to minimize the error.

$$\gamma(s) = p_n(s)u(s) - p(s)u_r(s) \quad (119)$$

The desired velocity response  $v_r(s)$  is

$$v_r(s) = p_n(s)u(s) \quad (120)$$

Substituting Eqn (118) into Eqn (120) we have

$$u = \frac{\dot{v}_r}{b} + \frac{c}{b}r \quad (121)$$

The actual velocity loop subjected to disturbance is

$$a = bu_r - cv + d \quad (122)$$

where  $a = \dot{v}$  is the actual acceleration and  $d$  is the lumped disturbances. System (122) is extended using the following state space representation [90]

$$\begin{cases} \dot{\gamma} = \dot{v}_r - a \\ \dot{\gamma} = ca - b\dot{u} - \dot{d} + \ddot{v}_r \end{cases} \quad (123)$$

Defining the following robust control

$$u_r = u + \frac{1}{b}[(G - c_n)\gamma - G_I \int_0^t S dt - G_S \int_0^t \text{sign}(S) dt] \quad (124)$$

where  $S = a - G\gamma - \dot{r}$ .  $G$ ,  $G_I$  and  $G_S$  are positive controller gains.

*Theorem 1:* For the extended system (123), the proposed robust controller (124) guarantees  $\gamma(t) \rightarrow 0$  as  $t \rightarrow \infty$ , under the following conditions:

1. The first order derivative of the lumped disturbance is bounded, i.e.,

$$\|\dot{d}\|_{\text{sup}} \leq d_{\text{sup}} \quad (125)$$

where  $d_{\text{sup}}$  is a positive scalar.

2. The controller gain satisfies

$$G_S > d_{\text{sup}} \quad (126)$$

Proof: Define the following Lyapunov candidate

$$V = \frac{1}{2}S^2 \quad (127)$$

Take the derivative of Eqn (127)

$$\begin{aligned} \dot{V} &= (a - G\gamma - \dot{v}_r)(\dot{a} - G\dot{\gamma} - \ddot{v}_r) \\ &= S(-ca + bu_r + \dot{d} - G\dot{\gamma} - \ddot{v}_r) \end{aligned} \quad (128)$$

Substituting Eqn (121) and (124) into Eqn (128) we can obtain

$$\begin{aligned} \dot{V} &= S(-G_I S - \text{sign}(S)G_S + \dot{d}) \\ &= -G_I S^2 - |S|G_S + S\dot{d} \end{aligned} \quad (129)$$

Substituting Eqn (126) into Eqn (129) we have

$$\dot{V} < 0 \quad (130)$$

and

$$\lim_{t \rightarrow \infty} a = G\gamma + \dot{r} \quad (131)$$

Substituting Eqn (131) into Eqn (123) yield

$$\dot{\gamma} = -G\gamma \quad (132)$$

Therefore the system is asymptotically stable, i.e.

$$\lim_{t \rightarrow \infty} (\gamma(t)) = 0 \quad (133)$$

QED

## 4.6 Experimental Validations

### 4.6.1 Contour Design

The desired contour is slightly modified from a plane dog bone contour by altering the z coordinate of each control point. The synthesized feedrate is  $20\text{mm/s}$  and the desired trajectory for each axis is illustrated in Figure 45.

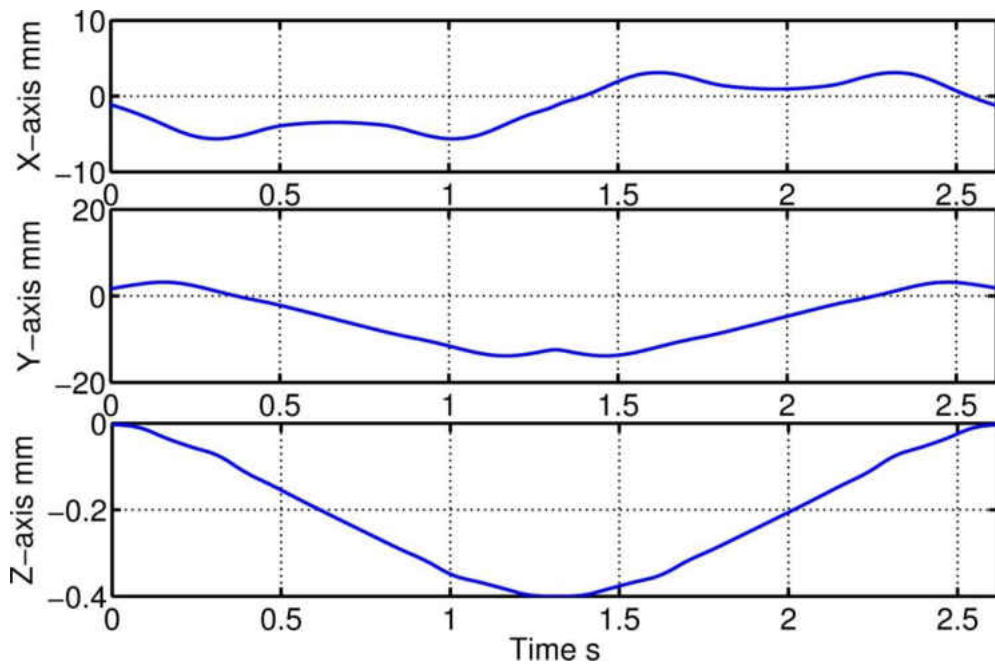


Figure 45: Reference trajectory for each feed drive

The performance difference between the coordinate control and decoupled control has been demonstrated in chapter 3. Therefore we will only compare the proposed controller to the task frame based controller [47].

#### 4.6.2 Experiment 1 Contour Following for Curves of Large Torsion

The first experiment is designed to compare the performance differences between the optimized and non-optimized moving frame. Since this experiment is designed to demonstrate the effectiveness of the optimized moving frame. Linear contour error estimation is used in accord with [47]. Our earlier simulation indicated the possibility of actuator saturations. To protect the motion system, the controller bandwidth was turned lower than normal level and the inner loop is disabled as well. The controller gains are listed in Table 7.

Table 7: Controller gains of contour control in  $\mathbb{R}^3$

$k_{pt}$	$k_{dt}$	$k_{pn}$	$k_{dn}$	$k_{pb}$	$k_{db}$
10000	100	150000	500	150000	500

Two series of frames were generated, the first series was generated using the moving trihedron [47] and the second one is obtained using the proposed optimization methods. The optimizing rotation angle  $\theta$  is calculated to minimize the cost function (99) and the results are shown in Figure 46. The cost functions before and after the optimizations are illustrated in Figure 47.



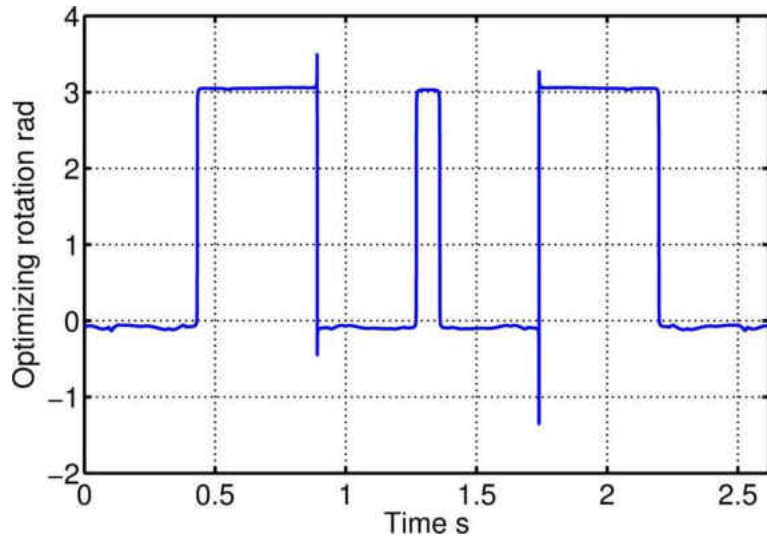


Figure 46: Optimizing rotation angle  $\theta$

By comparing Figure 46 and Figure 47, the original task frames were rotated by as much as 3 rad, indicating a non-smooth task frame for a smooth contour. While after the optimization, the maximum of cost function was reduced from 6.5 to less than  $4 \times 10^{-4}$ .

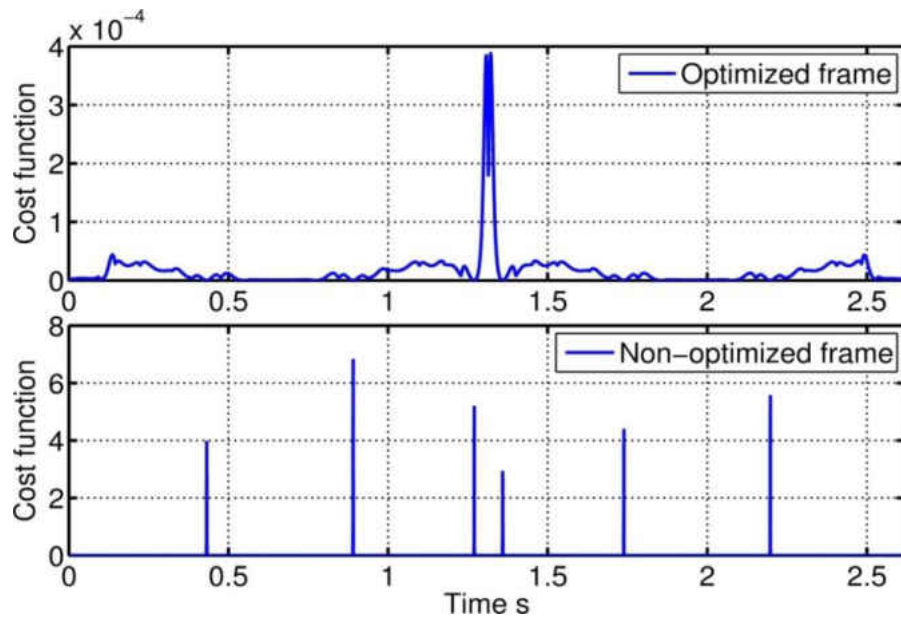


Figure 47: Cost functions of optimized and non-optimized move frames

The control signal based on the non-optimized moving frame is shown in Figure 48. Saturations happened in the control signal of all 3 axes (The control effort of *X,Y-axis* is within  $\pm 10V$ , corresponding to a maximum of  $3.5A$  winding current while the control effort of *Z-axis* corresponds to a  $5A$  winding current. Compare the cost function in Figure 47 and the saturation in Figure 48, saturations clearly happened when the transition of the local frame becomes drastic. Take an example of the saturation happened at 0.4315 second, the local frame flipped almost 180 degrees in just a couple of sampling intervals. As a result, the control system sensed a drastic change in the error signal and therefore generated an overtly large but unnecessary correction signal. It should be noted that the bandwidth of the 3 axes is lower than the normally level. A much more serious saturation is expected if the controller is at normal working condition. Saturations not only degrade the quality of the machined products but excess heat and high bus voltage punching the winding of the drive system. In addition, spikey control signals may cause high frequency structure dynamics, which are highly unwanted.

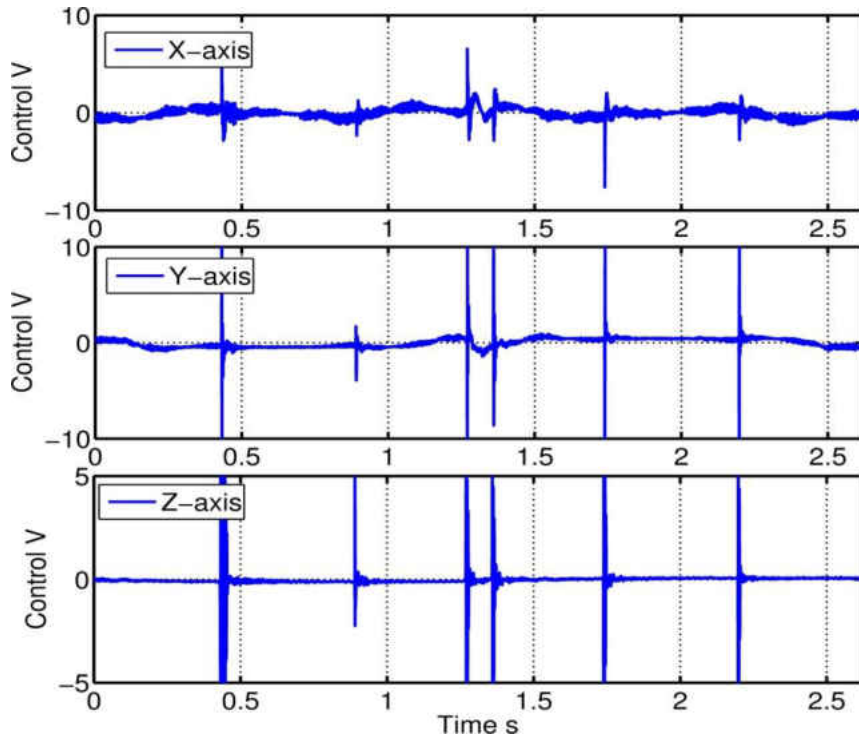


Figure 48: Control signal using non-optimized moving frame

The same experiment was repeated using the optimized moving frame and the control signal is in Figure 49. Compare to the non-optimized results, the proposed moving frame has no spikes in the control efforts. The contour error comparison is illustrated in Figure 50. At the place where no saturation happens, two frames show similar performance, which are expected. However, the optimized frame completely eliminates saturation and hence beats the non-optimized frame by considerable margins. Therefore the proposed frame is applicable to regular curve in  $\mathbb{R}^3$  while the non-optimized frame will fail at curves with large torsion.

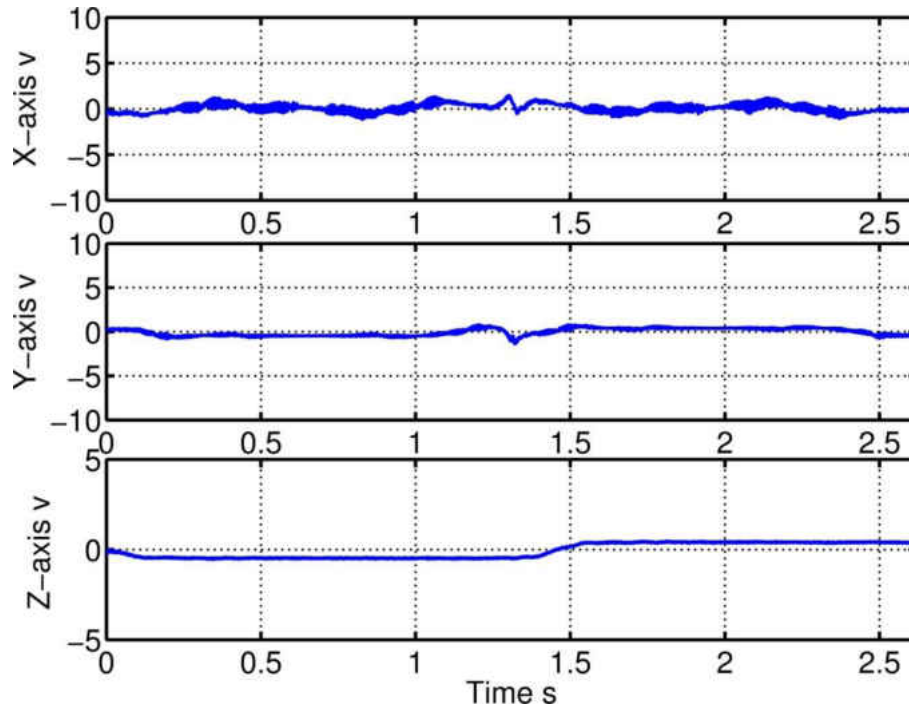


Figure 49: Control signal using optimized moving frame

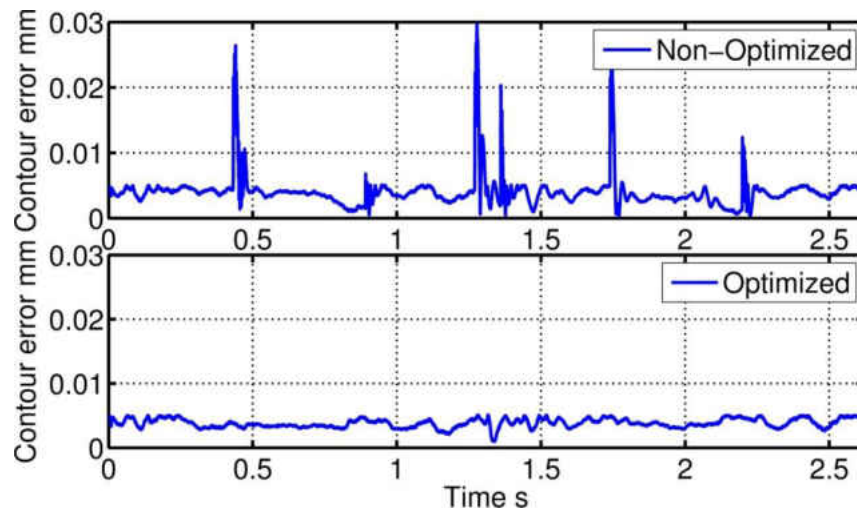


Figure 50: Performance comparison between the optimized and non-optimized frames

#### 4.6.3 Experiment 2 Performance of Full Controller

The dog bone contour experiment was repeated using the proposed contour error estimation method and the robust inner loop. The contour error is illustrated in Figure 51. From

Figure 51, the improved contour error estimation and robust inner loop significantly improve the contour following performance. The maximum contour error decreased from about  $5\mu\text{m}$  to less than  $1\mu\text{m}$ . The major contribution to this improvement is believed from the inner loop increasing the disturbance rejection capability of the system.

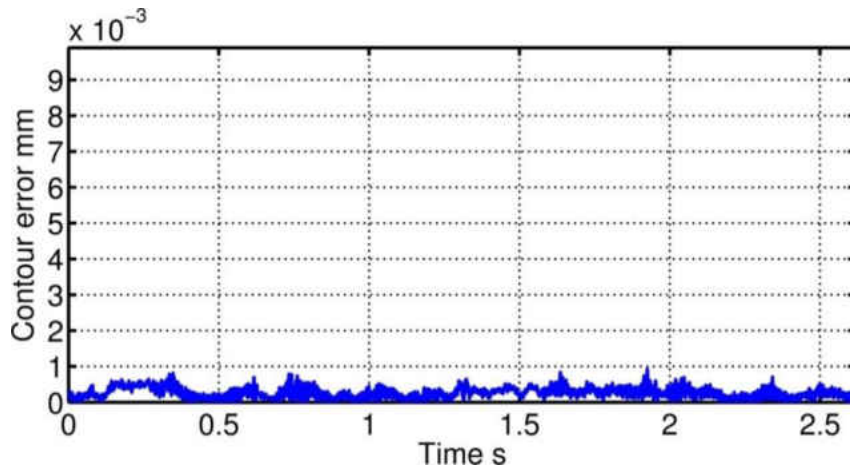


Figure 51: Contour error using robust inner loop and improved contour error model

#### 4.7 Summary

An optimization method is proposed in this chapter for contour following problem of regular curves in  $\mathbb{R}^3$ . The optimization method eliminates the unnecessary rotation of the local frame previously developed and thus avoids controller saturation. The effectiveness of the proposed method has been experimentally demonstrated. In addition, an improved contour error estimation and the robust inner loop were introduced to further improve the contour following performance of the control system. The proposed scheme can be readily applied to regular curves in  $\mathbb{R}^3$  even the torsion of the curve is significantly large for superior contour following performances.

## CHAPTER 5 FUTURE WORK

The first academic report of today's iterative learning control (ILC) can be dated back as early as 1978 [91]. Here the previous researches on ILC are briefly introduced. Unlike other learning controllers, like adaptive control changing the parameters of the controller, ILC only modifies the control signal. Repetitive control (RC) and ILC are based on similar principals except that RC is for continuous operations while ILC is suitable for discontinuous operations [92]. The major difference between RC and ILC can be explained by the setting of initial conditions. In RC, the new trial normally begins immediately after the previous trial ends and therefore the initial condition of RC is set to the ending condition of the previous trial. On the other hand, ILC typically starts with the same initial conditions. A simplified machining circle typically starts with the machine tools standing by until the workpiece to be cut is clamped. Then the machine tools start tracking the position commands from the position interpolator and returns to the initial position after completing the cutting process. The machine tools will rest at the same initial position until the machined workpiece is removed and the new workpiece is installed. From this description, the machining process is a discontinuous operation and ILC is the natural choice for performance improvement.

ILC is entirely based on using information from the previous trials to improve the performance of current trial. Therefore ILC is typically viewed as a feedforward control in the time domain and a feedback control in the iteration domain. As a result, ILC is seldom used as a standalone controller; instead it is typically designed as an add-on controller to the existing feedback control system. Traditionally, the focus of ILC is on improving the performance of systems executing a single, repeated operation, such as machine tools, industrial robot and

autonomous vehicles. Based on the CCC structure, an iterative learning contour controller has been proposed to reduce the contour errors over iterations [93]. This study employs variable cross coupling gains to calculate the contour error and the control signal from the ILC is directly inject into the motion system to reduce the contour error.

Nerveless, it should be noted that the performance and stability of ILC are still highly dependent of the non-repetitive disturbances and variations of system dynamics. Therefore, to achieve robustness and fast convergence, the stabilizing feedback controller should have the ability to compensate most non-repetitive disturbances. An  $\mathcal{L}1$  adaptive control based robust ILC has been proposed where the  $\mathcal{L}1$  adaptive control is designed as the inner feedback loop to compensate the low frequency, non-repetitive uncertainties [59, 60]. However, the research results are only demonstrated by computer simulations. Our future work will be focused on using the iterative learning control (ILC) [94] to improve the performance of multi-axis machine tools. This is of significant practical importance because in mass production industries, mechanical systems are often commanded to repeatedly perform the same tasks. For example, painter robot arms may paint the same spot of the same model for hundreds of times and the machine tools may keep cutting the same workwise day by day. The quality of the machined product can be further improved using the information obtained from the previous trials.

## LIST OF REFERENCES

- [1] E. Tung, "Identification and control of high-speed machine tools," PhD dissertation, University of California, Berkeley, 1993.
- [2] J. Liu, C. Xu and L. An, "Micro-machinable polymer-derived ceramic sensors for high-temperature applications," in SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring, pp. 7640. V1-7640. V10, 2010.
- [3] D. Bristow, "Iterative learning control for precision motion control of microscale and nanoscale tracking systems," PhD dissertation, University of Illinois at Urbana-Champaign, 2007.
- [4] Y. Koren, "Interpolator for a computer numerical control system," Computers, IEEE Transactions on, vol. 100, pp. 32-37, 1976.
- [5] D. Simon and C. Isik, "Optimal trigonometric robot joint trajectories," Robotica, vol. 9, pp. 379-386, 1991.
- [6] K. Erkorkmaz, "Optimal trajectory generation and precision tracking control for multi-axis machines," PhD dissertation, University of British Columbia, 2004.
- [7] Y. Altintas, Manufacturing automation: metal cutting mechanics, machine tool vibrations, and CNC design, Cambridge university press, 2012.



- [8] Y. Koren, Computer control of manufacturing systems, McGraw-Hill New York, 1983.
- [9] K. Erkorkmaz and Y. Altintas, "High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation," International Journal of Machine Tools and Manufacture, vol. 41, pp. 1323-1345, 2001.
- [10] M. Cheng, M. Tsai and J. Kuo, "Real-time NURBS command generators for CNC servo controllers," International Journal of Machine Tools and Manufacture, vol. 42, pp. 801-813, 2002.
- [11] M. Tomizuka, "Zero phase error tracking algorithm for digital control," Journal of Dynamic Systems, Measurement and Control, vol. 109, pp. 65-68, 1987.
- [12] S. Hara, Y. Yamamoto, T. Omata and M. Nakano, "Repetitive control system: a new type servo system for periodic exogenous signals," Automatic Control, IEEE Transactions on, vol. 33, pp. 659-668, 1988.
- [13] P. Van den Braembussche, J. Swevers and H. Van Brussel, "Linear motor ripple compensation using position-triggered repetitive control," in Atelier International IFAC Motion Control, pp. 385-389, 1998.

- [14] T. Lee, K. Tan, S. Lim and H. Dou, "Iterative learning control of permanent magnet linear motor with relay automatic tuning," *Mechatronics*, vol. 10, pp. 169-190, 2000.
- [15] G. Otten, T. Vries, J. Amerongen, A. Rankers and E. Gaal, "Linear motor motion control using a learning feedforward controller," *Mechatronics, IEEE/ASME Transactions on*, vol. 2, pp. 179-187, 1997.
- [16] C. Kempf and S. Kobayashi, "Disturbance observer and feedforward design for a high-speed direct-drive positioning table," *Control Systems Technology, IEEE Transactions on*, vol. 7, pp. 513-526, 1999.
- [17] A. Balestrino, A. Landi, M. Ould-Zmirli and L. Sani, "Automatic nonlinear auto-tuning method for Hammerstein modeling of electrical drives," *Industrial Electronics, IEEE Transactions on*, vol. 48, pp. 645-655, 2001.
- [18] 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.
- [19] K. Peng, B. Chen, G. Cheng and T. Lee, "Modeling and compensation of nonlinearities and friction in a micro hard disk drive servo system with nonlinear feedback control," *Control Systems Technology, IEEE Transactions on*, vol. 13, pp. 708-721, 2005.

- [20] C. Xu and R. Zhao, "Analytical modeling and experimental validation of force ripple and friction force for general direct drive systems," in International Manufacturing Science and Engineering Conference, pp. V002T02A024-V002T02A032, 2013.
- [21] Q. Xie, "Modeling and control of linear motor feed drives for grinding machines," PhD dissertation, Georgia Institute of Technology, 2008.
- [22] E. Tung, Y. Urushisaki, and M. Tomizuka, "Low velocity friction compensation for machine tool feed drives," in American Control Conference , pp. 1932-1936, 1993.
- [23] E. Tung and M. Tomizuka, "Feedforward tracking controller design based on the identification of low frequency dynamics," Journal of Dynamic Systems, Measurement and Control, vol. 115, pp. 348-356, 1993.
- [24] E. Tung, G. Anwar and M. Tomizuka, "Low velocity friction compensation and feedforward solution based on repetitive control," in American Control Conference, pp. 2615-2620, 1991.
- [25] Y. Koren and C. Lo, "Advanced controllers for feed drives," CIRP Annals-Manufacturing Technology, vol. 41, pp. 689-698, 1992.

- [26] S. Devasia, E. Eleftheriou and S. Moheimani, "A survey of control issues in nanopositioning," *Control Systems Technology, IEEE Transactions on*, vol. 15, pp. 802-823, 2007.
- [27] L. Tang and R. Landers, "Multiaxis contour control-the state of the art," *Control Systems Technology, IEEE Transactions on*, vol. 21, pp. 1997-2010, 2013.
- [28] Y. Koren and C. Lo, "Variable-gain cross-coupling controller for contouring," *CIRP Annals-Manufacturing Technology*, vol. 40, pp. 371-374, 1991.
- [29] Y. Shiuh and H. Lo, "Estimation of the contouring error vector for the cross-coupled control design," *Mechatronics, IEEE/ASME Transactions on*, vol. 7, pp. 44-51, 2002.
- [30] Y. Tarng, H. Chuang and W. Hsu, "An optimisation approach to the contour error control of CNC machine tools using genetic algorithms," *The International Journal of Advanced Manufacturing Technology*, vol. 13, pp. 359-366, 1997.
- [31] Y. Tarng, H. Chuang and W. Hsu, "Intelligent cross-coupled fuzzy feedrate controller design for CNC machine tools based on genetic algorithms,"

- International Journal of Machine Tools and Manufacture, vol. 39, pp. 1673-1692, 1999.
- [32] S. Yeh and P. Hsu, "A new approach to bi-axial cross-coupled control," in Control Applications. Proceedings of the 2000 IEEE International Conference on, pp. 168-173, 2000.
- [33] C. Brecher, D. Lindemann, M. Merz and C. Wenzel, "FPGA-based control system for highly dynamic axes in ultra-precision machining," in Proceedings of the ASPE Spring Topical Meeting, Boston, 2010.
- [34] S. Zschäck, J. Klöckner, I. Gushchina, A. Amthor, C. Ament and W. Fengler, "Control of nanopositioning and nanomeasuring machines with a modular FPGA based data processing system," Mechatronics, vol.23, pp. 257-263, 2013.
- [35] C. Krah and C. Klarenbach, "Fast and high precision motor control for high performance servo drives," in PCIM Conference, pp.326-333, 2010.
- [36] S. Chen, H. Liu and S. Ting, "Contouring control of biaxial systems based on polar coordinates," Mechatronics, IEEE/ASME Transactions on, vol. 7, pp. 329-345, 2002.

- [37] J. Yang and Z. Li, "A novel contour error estimation for position loop-based cross-coupled control," *Mechatronics, IEEE/ASME Transactions on*, vol. 16, pp. 643-655, 2011.
- [38] Y. Koren, "Cross-coupled biaxial computer control for manufacturing systems," *Journal of Dynamic Systems, Measurement and Control*, vol. 102, pp. 265-272, 1980.
- [39] Y. Shih, C. Chen and A. Lee, "A novel cross-coupling control design for bi-axis motion," *International Journal of Machine Tools and Manufacture*, vol. 42, pp. 1539-1548, 2002.
- [40] S. Yeh and P. Hsu, "Theory and applications of the robust cross-coupled control design," in *American Control Conference*, pp. 791-795, 1997.
- [41] Y. Altintas and B. Sencer, "High speed contouring control strategy for five-axis machine tools," *CIRP Annals-Manufacturing Technology*, vol. 59, pp. 417-420, 2010.
- [42] B. Sencer and Y. Altintas, "Modeling and control of contouring errors for five-axis machine tools—part ii: Precision contour controller design," *Journal of Manufacturing Science and Engineering*, vol. 131, pp. 031007.1-031007-10, 2009.

- [43] C. Lo, "Cross-coupling control of multi-axis manufacturing systems," PhD dissertation, University of Michigan, 1992.
- [44] S. Chen and K. Wu, "Contouring control of smooth paths for multiaxis motion systems based on equivalent errors," *Control Systems Technology, IEEE Transactions on*, vol. 15, pp. 1151-1158, 2007.
- [45] D. Sun and M. Tong, "A synchronization approach for the minimization of contouring errors of CNC machine tools," *Automation Science and Engineering, IEEE Transactions on*, vol. 6, pp. 720-729, 2009.
- [46] G. Chiu, "Contour tracking of machine tool feed drive systems," in *American Control Conference*, pp. 3833-3837, 1998.
- [47] G. Chiu and M. Tomizuka, "Contouring control of machine tool feed drive systems: A task coordinate frame approach," *Control Systems Technology, IEEE Transactions on*, vol. 9, pp. 130-139, 2001.
- [48] G. Chih, "Coordination control of multiple axes mechanical systems: theory and applications to machining and computer disk file systems," PhD dissertation, University of California, Berkeley 1994.
- [49] F. Iler, "Non-conventional machining of particle reinforced metal matrix composite," *International Journal of Machine Tools & Manufacture*, vol. 40, pp. 1351-1366, 2000.

- [50] S. Gordon and M. Hillery, "Development of a high-speed CNC cutting machine using linear motors," *Journal of Materials Processing Technology*, vol. 166, pp. 321-329, 2005.
- [51] K. Kinoshita, T. Sogo and N. Adachi, "Iterative learning control using adjoint systems and stable inversion," *Asian Journal of Control*, vol. 4, pp. 60-67, 2002.
- [52] T. Sogo, "Stable inversion for nonminimum phase sampled-data systems and its relation with the continuous-time counterpart," in *Decision and Control Proceedings of the 41st IEEE Conference on*, pp. 3730-3735. 2002.
- [53] T. Tsao and M. Tomizuka, "Adaptive zero phase error tracking algorithm for digital control," *Journal of Dynamic Systems, Measurement and Control*, vol. 109, pp. 349-354, 1987.
- [54] M. Norrlöf and S. Gunnarsson, "Time and frequency domain convergence properties in iterative learning control," *International Journal of Control*, vol. 75, pp. 1114-1126, 2002.
- [55] K. Moore, "Iterative learning control: an expository overview," in *Applied and Computational Control, Signals, and Circuits*, Springer, pp. 151-214, 1999.



- [56] R. Longman, "Iterative learning control and repetitive control for engineering practice," *International Journal of Control*, vol. 73, pp. 930-954, 2000.
- [57] D. Kim and S. Kim, "An iterative learning control method with application for CNC machine tools," *Industry Applications, IEEE Transactions on*, vol. 32, pp. 66-72, 1996.
- [58] K. Barton, S. Mishra and E. Xargay, "Robust iterative learning control: L1 adaptive feedback control in an ILC framework," in *American Control Conference*, pp. 3663-3668. 2011.
- [59] B. Altin and K. L. Barton, "L1 adaptive control in an iterative learning control framework for nan positioning," 2013.
- [60] B. Altin and K. Barton, "L1 adaptive control in an iterative learning control framework: Stability, robustness and design trade-offs," in *American Control Conference*, pp. 6697-6702, 2013.
- [61] S. Kouro, P. Lezana, M. Angulo and J. Rodriguez, "Multicarrier PWM with DC-link ripple feedforward compensation for multilevel inverters," *Power Electronics, IEEE Transactions on*, vol. 23, pp. 52-59, 2008.
- [62] C. Ru and L. Sun, "Improving positioning accuracy of piezoelectric actuators by feedforward hysteresis compensation based on a new

- mathematical model," *Review of Scientific Instruments*, vol. 76, pp. 09511.1-095111.8, 2005.
- [63] Z. Jamaludin, H. Van Brussel, and J. Swevers, "Quadrant glitch compensation using friction model-based feedforward and an inverse-model-based disturbance observer," in *Advanced Motion Control, 10th IEEE International Workshop on*, pp. 212-217, 2008.
- [64] M. Gafvert, "Dynamic model based friction compensation on the Furuta pendulum," in *Control Applications, Proceedings of the 1999 IEEE International Conference on*, pp. 1260-1265, 1999.
- [65] B. Armstrong-Hélouvry, P. Dupont and C. De Wit, "A survey of models, analysis tools and compensation methods for the control of machines with friction," *Automatica*, vol. 30, pp. 1083-1138, 1994.
- [66] C. Canudas de Wit, H. Olsson, K. Astrom and P. Lischinsky, "A new model for control of systems with friction," *Automatic Control, IEEE Transactions on*, vol. 40, pp. 419-425, 1995.
- [67] F. Al-Bender, V. Lampaert and J. Swevers, "The generalized Maxwell-slip model: a novel model for friction simulation and compensation," *Automatic Control, IEEE Transactions on*, vol. 50, pp. 1883-1887, 2005.

- [68] J. Xu and K. Abidi, "Discrete-time output integral sliding-mode control for a piezomotor-driven linear motion stage," *Industrial Electronics, IEEE Transactions on*, vol. 55, pp. 3917-3926, 2008.
- [69] J. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, pp. 308-313, 1965.
- [70] A. Dubi, *Monte Carlo applications in systems engineering*, John Wiley & Sons, 2000.
- [71] C. De Wit and P. Lischinsky, "Adaptive friction compensation with partially known dynamic friction model," *International Journal of Adaptive Control and Signal Processing*, vol. 11, pp. 65-80, 1997.
- [72] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Neural Networks, IEEE International Conference on*, pp. 1942-1948, 1995.
- [73] C. Röhrig and A. Jochheim, "Motion control of linear permanent magnet motors with force ripple compensation," in *Proceedings of the Third International Symposium on Linear Drives for Industry Applications*, pp. 478-483, 2001.
- [74] L. Lu, B. Yao, Q. Wang and Z. Chen, "Adaptive robust control of linear motors with dynamic friction compensation using modified LuGre model," *Automatica*, vol. 45, pp. 2890-2896, 2009.

- [75] Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Evolutionary Computation. Proceedings of the 2001 Congress on* , pp. 81-86, 2001.
- [76] K. Parsopoulos and M. Vrahatis, "Recent approaches to global optimization problems through particle swarm optimization," *Natural Computing*, vol. 1, pp. 235-306, 2002.
- [77] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *Evolutionary Computation, IEEE Transactions on*, vol. 6, pp. 182-197, 2002.
- [78] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *Evolutionary Computation, IEEE Transactions on*, vol. 6, pp. 58-73, 2002.
- [79] S. Phillips and K. Ballou, "Friction modeling and compensation for an industrial robot," *Journal of Robotic Systems*, vol. 10, pp. 947-971, 1993.
- [80] J. Hung, W. Gao and J. C. Hung, "Variable structure control: a survey," *Industrial Electronics, IEEE Transactions on*, vol. 40, pp. 2-22, 1993.
- [81] W. Gao and J. Hung, "Variable structure control of nonlinear systems: a new approach," *Industrial Electronics, IEEE Transactions on*, vol. 40, pp. 45-55, 1993.

- [82] B. Yao, S. Chan and D. Wang, "Unified formulation of variable structure control schemes for robot manipulators," *Automatic Control, IEEE Transactions on*, vol. 39, pp. 371-376, 1994.
- [83] W. Gao, Y. Wang and A. Homaifa, "Discrete-time variable structure control systems," *Industrial Electronics, IEEE Transactions on*, vol. 42, pp. 117-122, 1995.
- [84] S. Sarpturk and Y. Istefanopulos, "On the stability of discrete-time sliding mode control systems," *Automatic Control, IEEE Transactions on*, vol. 32, pp. 930-932, 1987.
- [85] A. Poo, J. Bollinger and G. Younkin, "Dynamic errors in type 1 contouring systems," *Industry Applications, IEEE Transactions on*, vol. IA-8, pp. 477-484, 1972.
- [86] L. Xu and B. Yao, "Coordinated adaptive robust contour tracking of linear-motor-driven tables in task space," in *Decision and Control. Proceedings of the 39th IEEE Conference on*, pp. 2430-2435, 2000.
- [87] H. Xiong, B. Yao and Q. Wang, "Coordinated adaptive robust contouring controller design for an industrial biaxial precision gantry," *Mechatronics, IEEE/ASME Transactions on*, vol. 15, pp. 728-735, 2010.

- [88] C. Peng and C. Chen, "Biaxial contouring control with friction dynamics using a contour index approach," *International Journal of Machine Tools and Manufacture*, vol. 47, pp. 1542-1555, 2007.
- [89] M. Do Carmo, *Differential geometry of curves and surfaces*, Prentice-Hall Englewood Cliffs, 1976.
- [90] C. Chen and R. Xu, "Tracking control of robot manipulator using sliding mode controller with performance robustness," *Journal of Dynamic systems, Measurement and Control*, vol. 121, pp. 64-70, 1999.
- [91] M. Uchiyama, "Formulation of high-speed motion pattern of a mechanical arm by trial," *Transactions SICE* , vol. 14, pp. 706-712, 1978.
- [92] G. Hillerström and K. Walgama, "Repetitive control theory and applications-a survey," in *Proceedings of the 13th IFAC World Congress*, pp. 1-6, 1996.
- [93] K. Barton and A. Alleyne, "A cross-coupled iterative learning control design for precision motion control," *Control Systems Technology, IEEE Transactions on*, vol. 16, pp. 1218-1231, 2008.
- [94] D. Bristow, M. Tharayil and A. Alleyne, "A survey of iterative learning control," *Control Systems, IEEE*, vol. 26, pp. 96-114, 2006.