

1-1-2012

# A Novel Position Domain Controller For Contour Tracking Performance Improvement

Truong Dam  
*Ryerson University*

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>



Part of the [Aerospace Engineering Commons](#)

---

## Recommended Citation

Dam, Truong, "A Novel Position Domain Controller For Contour Tracking Performance Improvement" (2012). *Theses and dissertations*. Paper 1309.

This Thesis is brought to you for free and open access by Digital Commons @ Ryerson. It has been accepted for inclusion in Theses and dissertations by an authorized administrator of Digital Commons @ Ryerson. For more information, please contact [bcameron@ryerson.ca](mailto:bcameron@ryerson.ca).

A NOVEL POSITION DOMAIN CONTROLLER FOR CONTOUR TRACKING  
PERFORMANCE IMPROVEMENT

by

Truong Dam

Bachelor of Aerospace Engineering with Honors

Ryerson University

Toronto, Ontario, Canada, 2010

A Thesis

presented to Ryerson University

in partial fulfillment of the  
requirements for the degree of

Master of Applied Science

in the Program of  
Aerospace Engineering

Toronto, Ontario, Canada, 2012

© Truong Dam 2012

## **AUTHOR'S ACKNOWLEDGEMENT**

### **AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS**

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

# **ABSTRACT**

## **A NOVEL POSITION DOMAIN CONTROLLER FOR CONTOUR TRACKING PERFORMANCE IMPROVEMENT**

Truong Dam

A thesis for the degree of

Master of Applied Science, 2012

Department of Aerospace Engineering, Ryerson University

A common problem with modern manufacturing processes that utilize high feed-rate machining is how to accurately track a given contour for the tool center point (TCP) of a system. Various methods have been developed to increase axial tracking performance and contouring performance of computerized numerical control (CNC) machines. These include: high gain feedback controllers, feedforward controllers, zero phase error tracking controllers (ZPETC), cross-coupled control (CCC), and iterative learning control to mention a few. The common factor amongst these methods is that they are all based in time domain. This thesis will propose a new control law based in position domain applied to contour tracking control of a CNC machine. The goal of this developed controller is to improve the overall tracking and contouring performance of a CNC system. The idea behind a position domain control involves transforming the dynamics of a system from time domain into position domain through a one-to-one mapping.

In the position domain system control, the motion of one of the axis is used as an independent reference by sampling equidistantly to control the remaining axes according to the contouring requirements. The overall contour error in a position domain controller should be lower relative to an equivalent time domain controller since there will be a zero tracking error from the reference motion. The stability of the proposed position domain control is proven through the Lyapunov method. Simulations with linear and nonlinear TCP contours using the proposed position domain controller and an equivalent time domain controller indicate that the proposed position domain control can improve tracking and contouring performance. In addition, a position domain controller with cross-coupled control was also proposed to further improve contour performance.

## **ACKNOWLEDGEMENT**

The first acknowledgement is to my supervisor, Dr. Ouyang, for his guidance throughout this project. It would not be possible without his knowledge, time, and dedication. The second acknowledgement is to the Aerospace Engineering faculty and staff at Ryerson University and the Ontario Graduate Scholarship, for their assistance so that I was able to pursue this opportunity.

A special acknowledgement to my parents for their care and support. And finally to my fiancée for her patience and understanding. All your encouragements were very much appreciated.

## **DEDICATION**

*To my parents*

# TABLE OF CONTENTS

Chapter 1 : Introduction .....	1
1.1 Background .....	1
1.2 Motivation and objective .....	6
1.3 Organization of Contents .....	7
Chapter 2 : Literature Reviews .....	8
2.1 Contour error estimation .....	11
2.2 Tracking error control .....	15
2.3 Contouring error control .....	22
2.4 Advanced control methods .....	26
2.5 Remarks .....	33
Chapter 3 : Position domain control system design .....	34
3.1 Dynamics model .....	34
3.2 Time domain control .....	35
3.3 Time domain control with cross-coupled control .....	36
3.4 Position domain control .....	37
3.5 Stability analysis .....	40
3.5.1 Preparation and Lemma .....	40
3.5.2 Notations and theorem .....	41
3.5.3 Proof of stability .....	43
3.6 Position domain control with cross-coupled control .....	48
Chapter 4 : Simulation Results .....	50

4.1 Simulation setup.....	50
4.2 Trajectory planning.....	51
4.3 Gain selection.....	60
4.4 Linear contour tracking results .....	64
4.4.1 Zigzag contour .....	64
4.4.2 Diamond Contour.....	68
4.5 Circular contour results.....	74
4.6 Other simulations .....	79
4.6.1 High controller gains.....	79
4.6.2 PDC equidistant sampling.....	86
4.7 Mismatching dynamics .....	88
4.8 Remarks .....	94
Chapter 5 : Conclusion and Discussion .....	96
5.1 General review .....	96
5.2 Main contributions .....	97
5.3 Future development .....	98
Appendix A:.....	99
A.1 Primary program .....	99
A.2 Secondary functions .....	102
Bibliography .....	106

## LIST OF TABLES

Table 1: Non-mismatching System dynamic parameters .....	50
Table 2: TDC sampling rate and PDC equidistant sampling.....	50
Table 3: Initial and final axial positions for linear and circular contour simulations ( <i>pi</i> and <i>pf</i> denote initial and final position respectively.).....	53
Table 4: Segment and total motion time .....	53
Table 5: Maximum absolute axial velocities .....	57
Table 6: Linear contour controller gains for TDC, PDC, and CCC .....	60
Table 7: Mean and standard deviation (S.D) for zigzag motion axial tracking error .....	64
Table 8: Magnitude of contour error mean, standard of deviation, and maximum .....	68
Table 9: Axial tracking error mean and standard deviation (S.D) for diamond motion.....	69
Table 10: Magnitude of contour error mean, standard of deviation, and maximum .....	73
Table 11: Axial tracking error mean and standard deviation (S.D) for circular motion.....	74
Table 12: Magnitude of contour error mean, standard of deviation, and maximum .....	78
Table 13: Higher control gains .....	79
Table 14: Contour errors with increased controller gains.....	84
Table 15: Mismatching system dynamic parameters.....	88
Table 16: Mean contour error with mismatching dynamics .....	88

## LIST OF FIGURES

Figure 1-1: Various types of CNC machines A) Robot B) Milling machine C) Turning machine D) Machining center E) Wire EDM F) FMS line (Suh, Kang, Chung, & Stroud, 2008) .....	2
Figure 1-2: A typical control loop implemented in CNC machining (Suh, Kang, Chung, & Stroud, 2008).....	3
Figure 1-3: A typical control loop of a contouring system (Suh, Kang, Chung, & Stroud, 2008) .	4
Figure 1-4: Hierarchical levels in CNC controllers (Suh, Kang, Chung, & Stroud, 2008) .....	4
Figure 2-1: Tracking error contouring error .....	8
Figure 2-2: Linear contour .....	9
Figure 2-3: Tracking error and contouring error (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991).....	10
Figure 2-4: Possible cases of contouring error (Erkorkmaz, Yeung, & Altintas, 2006).....	12
Figure 2-5: Contour error for Case 1 and Case 2 (Erkorkmaz, Yeung, & Altintas, 2006) .....	14
Figure 2-6: A conventional feedforward controller inverse to plant dynamics .....	16
Figure 2-7: Zero phase error tracking system (Tomizuka, 1987) .....	18
Figure 2-8: A feedforward and feedback control loop with IKF (Weck & Ye, 1990) .....	20
Figure 2-9: The control scheme proposed by Van Den Braembussche (Van Den Braembussche, Swevers, Van Brussel, & Vanherck, 1996) .....	21
Figure 2-10: Biaxial cross-coupled control structure (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991).....	23
Figure 2-11: Variable-gain cross coupled control (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991).....	24
Figure 2-12: Optimal coupled control(Srinivasan & Kulkarni, 1990).....	26

Figure 2-13: Iterative learning control structure (Barton & Alleyne, 2008).....	27
Figure 2-14: Structure of a learning-type PID controller (Kim & S., 1996) .....	29
Figure 2-15: ILC controller scheme with a feedback loop (Barton & Alleyne, 2008).....	30
Figure 2-16: An overall structure of and ILC system (Bristow, Tharayil, & Alleyne, 2006) .....	30
Figure 2-17: Iterative learning control with cross-coupled control(Barton & Alleyne, 2008).....	31
Figure 2-18: Repetitive control diagram (Tan, Chua, Zhao, Yang, & M.T, 2009) .....	32
Figure 3-1: Schematic of a CNC machine .....	34
Figure 4-1: Simulation outline .....	51
Figure 4-2: Desired trajectories for zigzag contour axial positions and velocities.....	54
Figure 4-3: Desired trajectories for diamond contour axial positions and velocities .....	55
Figure 4-4: Desired trajectories for circular contour axial positions and velocities .....	56
Figure 4-5: Desired (A) Zigzag, (B) Diamond and (C) Circular contours in planar views .....	58
Figure 4-6: Desired (A) Zigzag, (B) Diamond and (C) Circular contours in 3D views .....	59
Figure 4-7: Zigzag contour tracking error variations with increasing controller gains .....	61
Figure 4-8: Diamond contour tracking error variations with increasing controller gains .....	62
Figure 4-9: Circular contour tracking error variations with increasing controller gains .....	63
Figure 4-10: Zigzag contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC .....	65
Figure 4-11: Zigzag motion contour error magnitude for TDC, TDC with CCC, PDC, and PDC with CCC .....	66
Figure 4-12: Zigzag contour tracking of TDC, TDC with CCC, PDC, and PDC with CCC .....	67
Figure 4-13: Diamond contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC.....	70

Figure 4-14: Diamond motion contour error magnitude for TDC, TDC with CCC, PDC, and PDC with CCC.....	71
Figure 4-15: Diamond contour tracking of TDC, TDC with CCC, PDC, and PDC with CCC....	72
Figure 4-16: Circular contour tracking error for TDC, PDC, and PDC with CCC .....	75
Figure 4-17: Circular motion contour error for TDC, PDC, and PDC with CCC .....	76
Figure 4-18: Circular contour tracking of TDC, TDC with CCC, PDC, and PDC with CCC .....	77
Figure 4-19: Zigzag contour axial tracking errors with increased controller gains .....	80
Figure 4-20: Diamond contour axial tracking errors with increased controller gains .....	81
Figure 4-21: Circular contour axial tracking errors with increased controller gains .....	82
Figure 4-22: Mean contouring error with increasing controller gains .....	83
Figure 4-23: Contouring performance of PDC with increasing CCC gains .....	85
Figure 4-24: Contouring performance of PDC with varying sampling distance .....	86
Figure 4-25: Zigzag contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC .....	89
Figure 4-26: Diamond contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC.....	90
Figure 4-27: Circular contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC .....	91
Figure 4-28: Contouring error for (a) zigzag (b) diamond (c) circular contours with mismatching dynamics .....	93

## ACRONYMS

CCC	<i>Cross-Coupled Control</i>
CCILC	<i>Cross-Coupled Iterative Learning Control</i>
CNC	<i>Computer Numerical Control</i>
EDC	<i>Event Driven Control</i>
GPC	<i>Generalized Predictive Control</i>
ILC	<i>Iterative Learning Control</i>
MTA	<i>Multi-axis</i>
PAA	<i>Parameter Adaptation Algorithm</i>
PD	<i>Proportional Derivative</i>
PDC	<i>Position Domain Control</i>
PI	<i>Proportional Integral Control</i>
PID	<i>Proportional Integral Derivative</i>
PTC	<i>Perfect Tracking Control</i>
PTP	<i>Point-To-Point</i>
RC	<i>Repetitive control</i>
TCP	<i>Tool center point</i>
TDC	<i>Time Domain Control</i>
ZPETC	<i>Zero Phase Error Tracking Control</i>
NC	<i>Numerical Control</i>

## LIST OF SYMBOLS

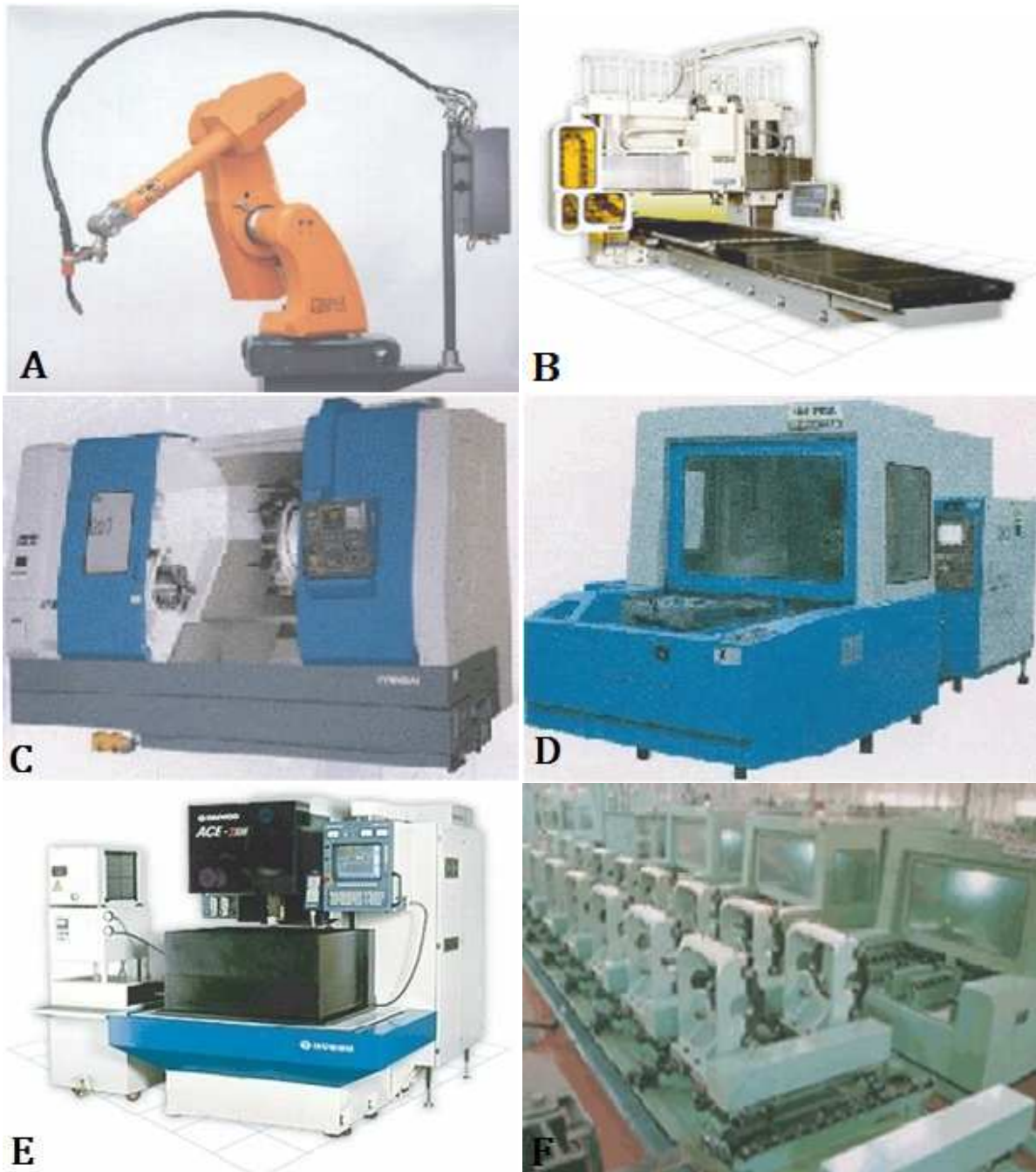
$\ddot{q}_i$	<i>Axial acceleration</i>
$\dot{q}_i$	<i>Axial velocity</i>
$T_i$	<i>Control input</i>
$c_i$	<i>Damping</i>
$k_i$	<i>Spring constant</i>
$m_i$	<i>Mass</i>
$q_i$	<i>Axial position</i>
$q_i'$	<i>Relative velocity</i>
$q_i''$	<i>Relative acceleration</i>
$P$	<i>Actual position</i>
$R$	<i>Reference position</i>
$e$	<i>Tracking error</i>
$\theta$	<i>Angle between the x-axis and the line connecting the origin with the reference point</i>
$\epsilon$	<i>Contouring error</i>

# Chapter 1 : INTRODUCTION

## 1.1 BACKGROUND

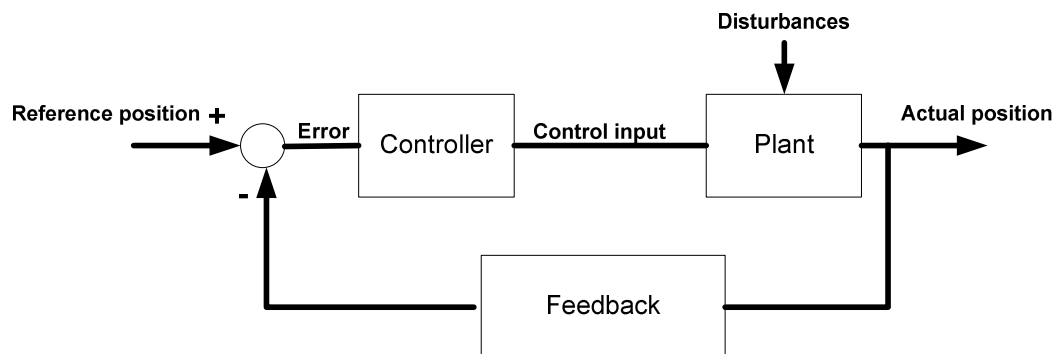
Computer numerical control (CNC) machines are a class of automated machine tools first developed during the 1950s. Their purpose was to aid in the machining of components with complex geometry at a greater rate and accuracy according to design specifications. The first generation of CNC machines, known at the time as numerical control (NC), was developed by John Parsons and the MIT Servomechanism Laboratory in the late 1940s and early 1950s (Suh, Kang, Chung, & Stroud, 2008). At that time, these machines were very large with intricate mechanical linkages requiring punch cards to operate. Therefore, they did not offer much in terms feasibility for general manufacturing. The first significant breakthrough for NC development occurred in the 1970s with the incorporation of computers into the control system of these machines. Thus, the technology became known as CNC. This eventually led to its vast acceptance in the manufacturing sector as well as garnering interests from researchers to improve the overall performance of systems (Koren Y. , 1983).

Prior to the introduction of NC machines, machined parts were manufactured by manually operated machines and for the most part, the geometric accuracy of these parts were largely determined by the skill and experience of an operator. The dimensional quality of these parts was periodically checked for conformity and the settings on the machines were tweaked to compensate for any geometric errors. This method of manufacturing for mass production has long been obsolete with the introduction of CNC machines. Today, there are a variety of CNC machines as shown in Figure 1-1, each with a particular application.



**Figure 1-1: Various types of CNC machines A) Robot B) Milling machine C) Turning machine D) Machining center E) Wire EDM F) FMS line (Suh, Kang, Chung, & Stroud, 2008)**

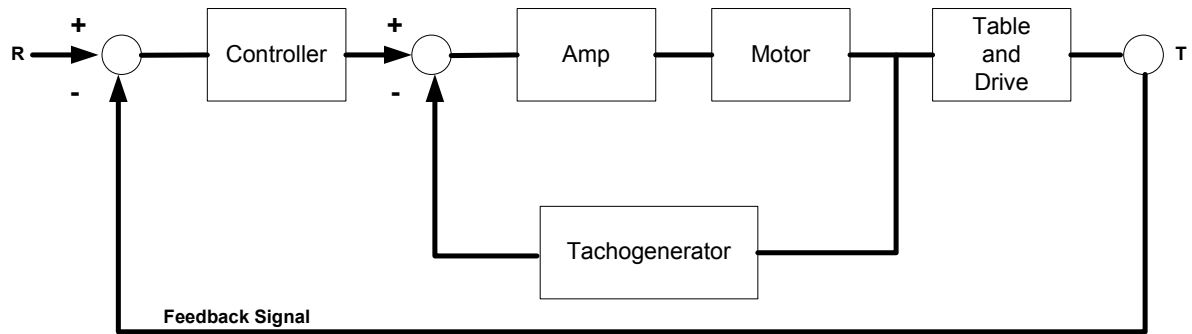
Generally speaking, there are two basic modes of operation for CNC systems: the first is the point-to-point (PTP) type, and the second is the contouring type. In a PTP system, it is satisfactory, in terms of machine accuracy, to have good axial tracking to the final desired position. This can be achieved by a feedback control system with a structure shown in Figure 1-2.



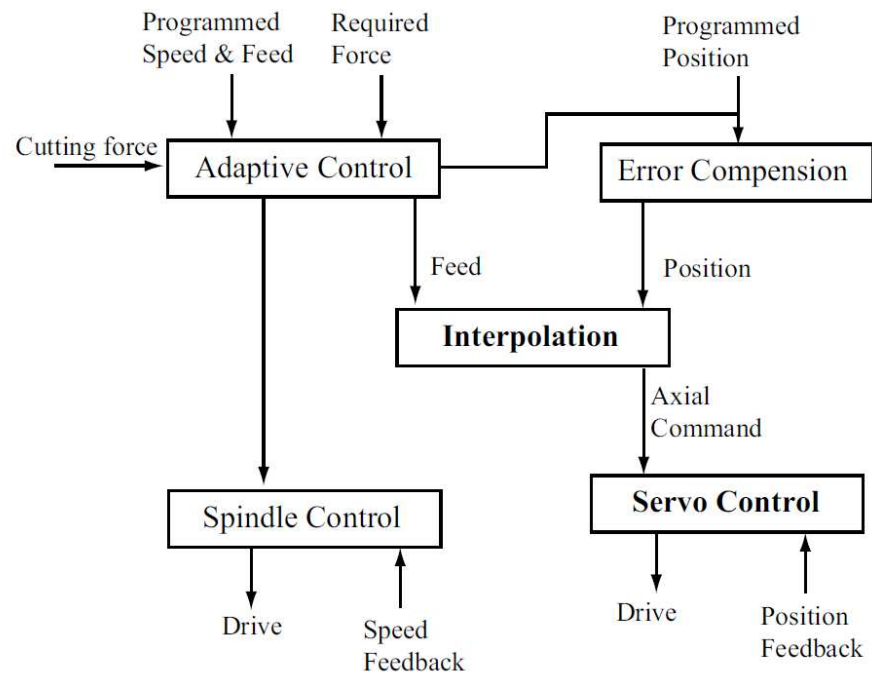
**Figure 1-2: A typical control loop implemented in CNC machining (Suh, Kang, Chung, & Stroud, 2008)**

For a contouring system, implementation of the control system is more complex with a number of different control strategies applicable to different types of applications. The complexity of the contouring type control has to do with the fact that the TCP needs to track a contour through synchronous motions between all of its axes. This is difficult to accomplish as the dynamics and disturbances of each axis are typically nonlinear and are inevitably coupled. In multiple degree of freedom systems, this becomes increasingly difficult to accurately model and therefore controlled.

In a conventional CNC machine control system, the motion of each axis is controlled independently of each other and each will typically have a control loop shown in Figure 1-3. The overall hierarchical level in a typical CNC controller is shown in Figure 1-4.



**Figure 1-3: A typical control loop of a contouring system (Suh, Kang, Chung, & Stroud, 2008)**



**Figure 1-4: Hierarchical levels in CNC controllers (Suh, Kang, Chung, & Stroud, 2008)**

Regardless of how well it is designed or built, all CNC machines will exhibit some form of inaccuracies. The main factors that affect these inaccuracies are classified as follows (Koren Y. , Control of Machine Tools, 1997):

1. Mechanical hardware deficiencies
2. Cutting process effects
3. Controller and drive dynamics

Errors contributed to the mechanical deficiencies are the results of backlash as well as errors from the geometric inconsistencies during the manufacturing and assembly of the machine. Cutting process errors occur during the machining of the part itself. During the cutting process, force is applied to the cutting tool, which to a certain extent, will lead to tool deflection and degradation and result in inaccurate cuts to the final product. The final factor is related to the actuation system of the machine and particularly, the control algorithms used to control the motion of the system. All these factors are important and should be considered in the design and operation of CNC machines.

The scope of this thesis involves studying the errors derived from the implementation of control laws through the use of various control algorithms. This is currently perhaps the most crucial area of research since the resulting errors in the machining process can be directly correlated to the speed of the operation. With an increasing trend towards the utilization of higher machining speeds and higher feed rates, a greater emphasis is placed on improving the accuracy of the machine from a control perspective.

## 1.2 MOTIVATION AND OBJECTIVE

The motivation behind this research is to improve the individual axial tracking of CNC machines and to ultimately reduce the contouring error of the manufactured products. This thesis will propose a new control law based in position domain to reduce axial tracking errors of a multi-axial CNC system. As a result of reducing the tracking error, the overall contouring performance of the system will be improved.

The overall goals of the proposed position domain controller are:

1. To provide an alternative to time domain controllers
2. To obtain better contouring performance through improved tracking performance
3. To reduce the system cost for the master axis actuation
4. To eliminate the restriction of traditional equally sampling time systems
5. To simplify the implementation of feedback controllers

In this thesis, the proposed position domain control (PDC) and a time domain control (TDC) will be simulated for linear and non-linear contours and compared to demonstrate the performance improvements of the proposed position domain controller over the conventional time domain controllers. To fulfill these goals, the following objectives will be achieved:

1. Model the dynamics of a simplified CNC system in position domain
2. Formulate a position domain controller
3. Conduct a stability analysis of the proposed controller using the Lyapunov method
4. Perform a performance comparison between the proposed position domain controller and a conventional time domain controller

### 1.3 ORGANIZATION OF CONTENTS

The remaining contents of this thesis will be organized as follows:

#### **Chapter 2: Literary review**

Chapter 2 is a literary review of various approaches to improve the performance of CNC machines. It will review the early control strategies aimed at improving axial tracking errors and continue with the current methods focusing on the overall contouring performance. The chapter will also review some advanced control methods such as iterative learning control, repetitive control, and event-driven control.

#### **Chapter 3: Control system design**

Chapter 3 provides a mathematical description of the CNC system dynamics in position domain and introduces the proposed position domain control (PDC) scheme. A stability analysis for the proposed position domain control using the Lyapunov method is also provided.

#### **Chapter 4: Simulation results**

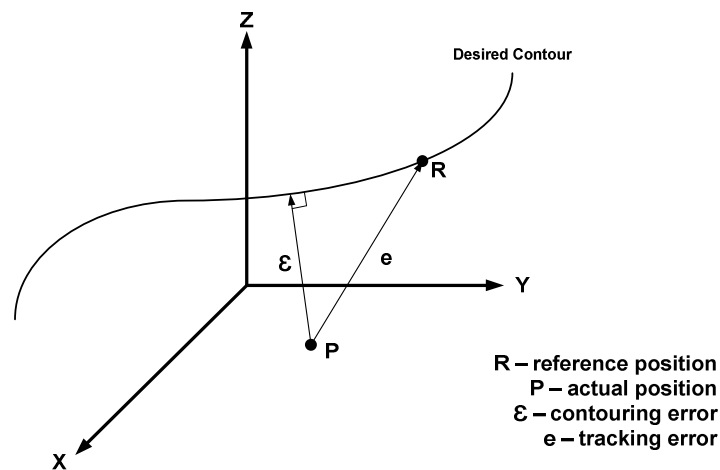
Simulation results for TDC and PDC controllers are presented and compared in terms of axial tracking and overall contouring performance for linear and nonlinear contours.

#### **Chapter 5: Discussion and conclusion**

A summary of the thesis is given in this chapter. It will conclude and discuss the findings of the simulation results as well as proposals for possible future developments.

## Chapter 2 : LITERATURE REVIEWS

Two common problems with high precision and high speed manufacturing processes involving CNC machines are the axial tracking and contouring performance of the TCP. For a given controller, these two types of errors, namely tracking error and contouring error, may exist within an acceptable range at low axial feed rate and also low system dynamics variations. However, for systems utilizing higher feed rates, the range of the error can become significantly larger as there is a tendency for the error to be proportional to the feed rate (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991). Therefore, consideration of these errors is important to obtain accurate CNC machining. As stated in Chapter 1, various factors can adversely influence the performance of a CNC machine resulting in inaccurately machined parts. Two main types of errors that can be generated are tracking error ( $e$ ) and contouring error ( $\epsilon$ ), which are displayed in Figure 2-1.



**Figure 2-1: Tracking error contouring error**

Tracking error results from an individual axial tracking performance, or how well the controller of each axis can track the respectively desired reference position and/or velocity. Mathematically, the tracking error can be expressed as,

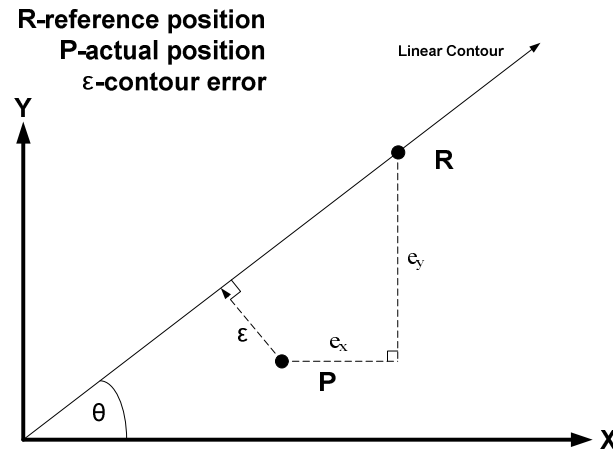
$$e = r - a \quad (2.1)$$

where  $r$  and  $a$  are the reference and actual positions, respectively.

Contouring error, on the other hand, is the error component orthogonal to the desired path. For a 2D planar contour, the contouring error can be expressed as,

$$\epsilon = -e_x \sin \theta + e_y \cos \theta \quad (2.2)$$

where  $e_x$  and  $e_y$  are the individual tracking errors of the  $x$ -axis and  $y$ -axis respectively.  $\theta$  is the angle from the  $x$ -axis to the line connecting the origin and the reference position  $R$  as shown in Figure 2-2. For linear contours,  $\theta$  is constant.



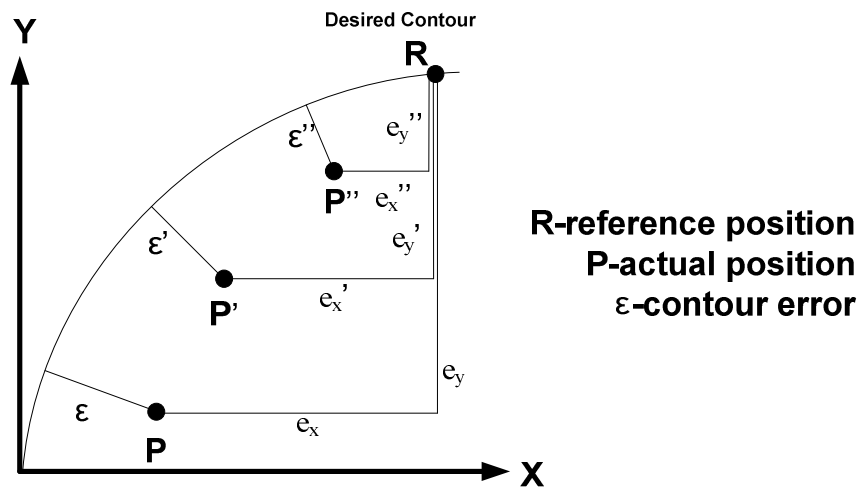
**Figure 2-2: Linear contour**

Eq. (2.2) applies only for linear contours. For non-linear contours such as circles, the contouring error is given as,

$$\epsilon = \sqrt{(R \sin \theta - e_x)^2 + (-R \cos \theta - e_y)^2} - R \quad (2.3)$$

where  $R$  is the radius of curvature. Note that in the circular contour,  $\theta$  is no longer constant and it changes depending on the reference position.

From the above definitions, it can be seen that the contouring error can be considered as a combination of all axial tracking errors in a multi-axis system. Various researches (Koren Y. , Cross-coupled biaxial computer control for manufacturing systems, 1980)(Fang & Chen, 2002)(Barton & Alleyne, 2008)(Hu, Yao, & Wang, 2009) have indicated that although advancements have been made to improve tracking performance, this does not necessarily guarantee good contouring performance. An example of this conclusion is shown in Figure 2-3. From this figure, the axial tracking errors ( $e_x$  and  $e_y$ ) have been reduced significantly from actual positions at point  $P$  to point  $P''$ , however, the change in the contouring error from  $\epsilon$  to  $\epsilon''$  remains relatively small.



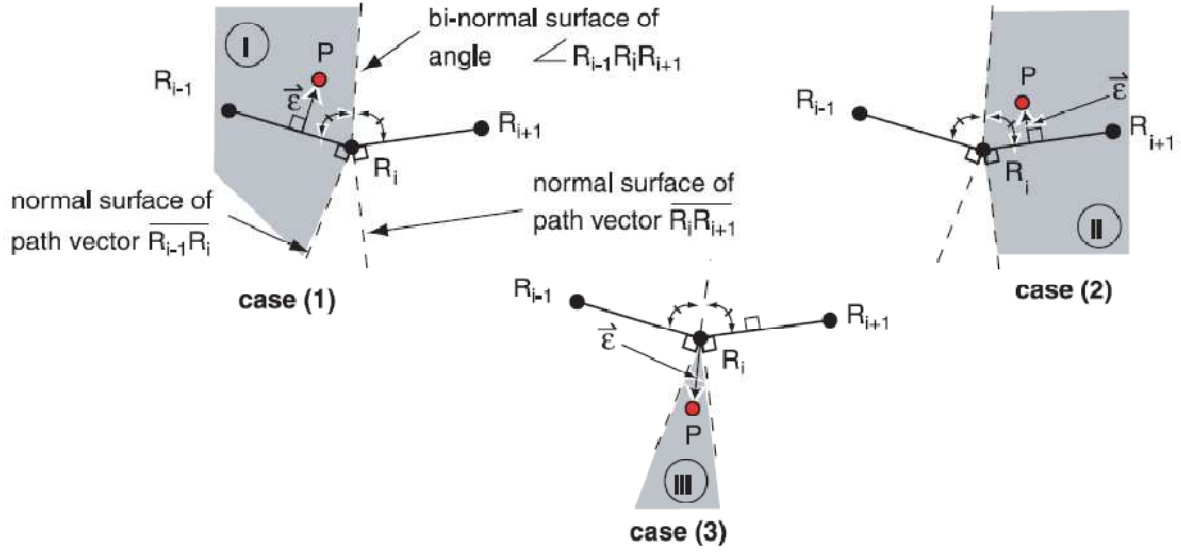
**Figure 2-3: Tracking error and contouring error (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991)**

Therefore, the overall contouring performance of a CNC system is usually taken as a performance index rather than the individual axial tracking performance. Such is the case, the

development to improve overall accuracy of CNC machines over the years have tended to focus on two areas of research. The first area of focus is on tracking control to improve axial tracking performance, while the second area of focus is on the overall contouring performance of the system (Ramesh, Mannan, & Poo, 2005). Efforts to improve tracking performance generally involve modification to the conventional feedback proportional (P), proportional-integral (PI), or proportional-integral-derivative (PID) controllers with the addition of a feedforward loop. Most of the axial control schemes presented in this review are variations of the feedforward control principle. On the other hand, the majority of the control schemes used to improve contouring performance are relatively limited in literature. The attempts that have been made are closely related to the concept of cross-coupled control or CCC introduced by Koren (Koren Y. , Cross-coupled biaxial computer control for manufacturing systems, 1980).

## **2.1 CONTOUR ERROR ESTIMATION**

Calculating the contour error of the TCP in real-time for 3D dimensional contour is highly difficult. Therefore it is often sufficient to estimate this error with the available system knowledge such as actual and reference TCP positions. The following derivation for contour error estimation is taken from (Erkorkmaz, Yeung, & Altintas, 2006).



**Figure 2-4: Possible cases of contouring error (Erkorkmaz, Yeung, & Altintas, 2006)**

From Figure 2-4,  $R_i = (x_i, y_i, z_i)$  represents the reference position while  $P = (x_p, y_p, z_p)$  represents the actual position. This indicates that there are three possible cases in estimating the contour error.

- Case 1: The contour error is estimated using the previous reference path segment  $\overrightarrow{R_{i-1}R_i}$ .
- Case 2: The contour error is estimated using the next reference path segment  $\overrightarrow{R_iR_{i+1}}$ .
- Case 3: The contour error is estimated using the closest reference point  $R_i$ .

To determine the estimate of the contour error, it is first necessary to define some definitions.

The normalized path segment vectors are given as,

$$\begin{aligned}\hat{\mathbf{r}}_i &= \frac{\overrightarrow{R_{i-1}R_i}}{\|\overrightarrow{R_{i-1}R_i}\|} = \frac{(x_i - x_{i-1})\mathbf{i} + (y_i - y_{i-1})\mathbf{j} + (z_i - z_{i-1})\mathbf{k}}{\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2}} \\ &= r_{x,i} \cdot \mathbf{i} + r_{y,i} \cdot \mathbf{j} + r_{z,i} \cdot \mathbf{k}\end{aligned}\quad (2.4)$$

And,

$$\begin{aligned}\hat{\mathbf{r}}_{i+1} &= \frac{\overrightarrow{R_iR_{i+1}}}{\|\overrightarrow{R_iR_{i+1}}\|} = \frac{(x_{i+1} - x_i)\mathbf{i} + (y_{i+1} - y_i)\mathbf{j} + (z_{i+1} - z_i)\mathbf{k}}{\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}} \\ &= r_{x,i+1} \cdot \mathbf{i} + r_{y,i+1} \cdot \mathbf{j} + r_{z,i+1} \cdot \mathbf{k}\end{aligned}\quad (2.5)$$

The position vector is defined as,

$$\begin{aligned}\mathbf{p} &= \overrightarrow{R_iP} = (x_p - x_i)\mathbf{i} + (y_p - y_i)\mathbf{j} + (z_p - z_i)\mathbf{k} \\ &= p_x \cdot \mathbf{i} + p_y \cdot \mathbf{j} + p_z \cdot \mathbf{k}\end{aligned}\quad (2.6)$$

The normal surfaces,  $N_i$  and  $N_{i+1}$ , which are perpendicular to the path segment vectors are given as,

$$N_i = r_{x,i} \cdot (x - x_i) + r_{y,i} \cdot (y - y_i) + r_{z,i} \cdot (z - z_i) = 0 \quad (2.7)$$

And,

$$N_{i+1} = r_{x,i+1} \cdot (x - x_i) + r_{y,i+1} \cdot (y - y_i) + r_{z,i+1} \cdot (z - z_i) = 0 \quad (2.8)$$

The bi-normal surface  $B_i$  for the angle  $\angle R_{i-1}R_iR_{i+1}$  is defined as,

$$\begin{aligned}B_i &= (r_{x,i} + r_{x,i+1}) \cdot (x - x_i) + (r_{y,i} + r_{y,i+1}) \times (y - y_i) \\ &\quad + (r_{z,i} + r_{z,i+1}) \cdot (z - z_i) = 0\end{aligned}\quad (2.9)$$

With the definitions above, it is now possible to determine the appropriate case with which to estimate the contour error.

If the angle  $\angle R_{i-1}R_iR_{i+1} \neq 180^\circ$  then,

$$\begin{cases} \text{Case 1} \rightarrow \text{if } B_i(x_p, y_p, z_p) < 0 \text{ and } N_i(x_p, y_p, z_p) \leq 0 \\ \text{Case 2} \rightarrow \text{if } B_i(x_p, y_p, z_p) \geq 0 \text{ and } N_{i+1}(x_p, y_p, z_p) \geq 0 \\ \text{Case 3} \rightarrow \text{if } N_i(x_p, y_p, z_p) > 0 \text{ and } N_{i+1}(x_p, y_p, z_p) < 0 \end{cases} \quad (2.10)$$

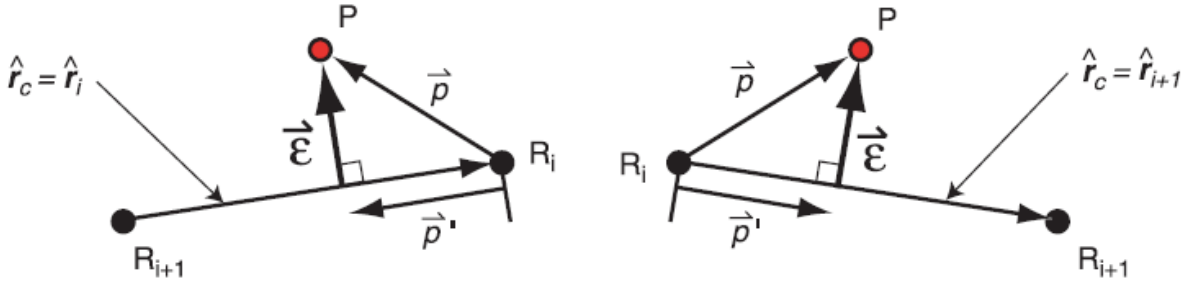
Otherwise if  $\angle R_{i-1}R_iR_{i+1} = 180^\circ$  then,

$$\begin{cases} \text{Case 3} \rightarrow \text{if } N_i(x_p, y_p, z_p) > 0 \text{ and } N_{i+1}(x_p, y_p, z_p) < 0 \\ \text{Case 3} \rightarrow \text{otherwise} \end{cases} \quad (2.11)$$

For Case 1 and Case 2, the projection vector  $\mathbf{p}'$  of the position vector  $\mathbf{p}$ , shown in Figure 2-5, is defined as,

$$\mathbf{p}' = (r_{xc}p_x + r_{yc}p_y + r_{zc}p_z)(r_{xc} \cdot \mathbf{i} + r_{yc} \cdot \mathbf{j} + r_{zc} \cdot \mathbf{k}) \quad (2.12)$$

Where  $r_{xc}$ ,  $r_{yc}$ , and  $r_{zc}$  are the components of the vector  $\mathbf{r}_c$ . Vector is defined as  $\mathbf{r}_c = \mathbf{r}_i$  or  $\mathbf{r}_c = \mathbf{r}_{i+1}$  for Case 1 and Case 2 respectively.



**Figure 2-5: Contour error for Case 1 and Case 2 (Erkorkmaz, Yeung, & Altintas, 2006)**

Therefore, the estimation of the contour error is given as,

$$\boldsymbol{\epsilon} = \mathbf{p} - \mathbf{p}' = \epsilon_x \cdot \mathbf{i} + \epsilon_y \cdot \mathbf{j} + \epsilon_z \cdot \mathbf{k} \quad (2.13)$$

The magnitude of the contour error is given as,

$$\|\epsilon\| = \sqrt{\epsilon_x^2 + \epsilon_y^2 + \epsilon_z^2} \quad (2.14)$$

For Case 3, the contour error vector is estimated by the vector  $\overrightarrow{R_l P}$ ,

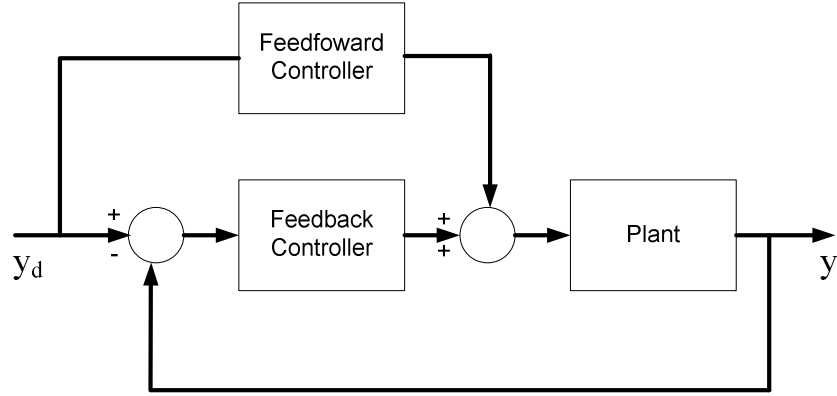
$$\epsilon = \epsilon_x \cdot \mathbf{i} + \epsilon_y \cdot \mathbf{j} + \epsilon_z \cdot \mathbf{k} \approx \overrightarrow{R_l P} = p_x \cdot \mathbf{i} + p_y \cdot \mathbf{j} + p_z \cdot \mathbf{k} \quad (2.15)$$

The magnitude of the contour error is given as,

$$\|\epsilon\| = \sqrt{p_x^2 + p_y^2 + p_z^2} \quad (2.16)$$

## 2.2 TRACKING ERROR CONTROL

The use of a standard PID feedback controller, also P or PI feedback controllers, to control each axis of a multi-axis system independently is highly common with traditional machine tooling control. However, there are some drawbacks when utilizing feedback controllers for high speed machining. With high speed machining there will be poor tracking of sharp edges as well as nonlinear contours. In addition, with the use of high control gains, there will be overshoot in the system. For the feedback controllers to be effective, it must also be finely tuned according to the dynamics of a particular system. These dynamics are not constant, particularly over time and will vary depending on working conditions such as, the mass of the work piece and the maintenance condition of the machine. To obtain a higher tracking accuracy from the controller, it is therefore necessary to develop more sophisticated axial control algorithms to overcome these limitations of the conventional feedback controllers. A common modification to the traditional feedback controller scheme is the addition of a feedforward controller to the control loop shown in Figure 2-6.



**Figure 2-6: A conventional feedforward controller inverse to plant dynamics**

One of the early efforts to improve axial tracking performance by adding a feedforward controller is called Perfect Tracking Controller (PTC). In PTC, the modeling of the plant dynamics is assumed to be highly accurate and also the information regarding the future reference trajectories is available to the controller. Due to the assumption of the highly accurate dynamics model of the drive system, the dynamics of this feedforward controller would essentially be the inverse of the closed-loop plant or close-loop system dynamics. The advantage of this process is an overall unity transfer function meaning the actual position would be equal to the required position. The PTC scheme would exhibit the following closed transfer function in discrete time as,

$$G(z^{-1}) = \frac{z^{-d}B_c(z^{-1})}{A_c(z^{-1})} \quad (2.17)$$

where  $z^{-d}$  represents a  $d$ -step delay caused by the delay in the plant and  $B_c(z^{-1}) = b_{c0} + b_{c1}z^{-1} + \dots + b_{cm}z^{-m}$ ,  $b_{c0} \neq 0$  and  $A_c(z^{-1}) = 1 + a_{c1}z^{-1} + \dots + a_{cn}z^{-n}$ . The output of the closed loop transfer function of the system are given as

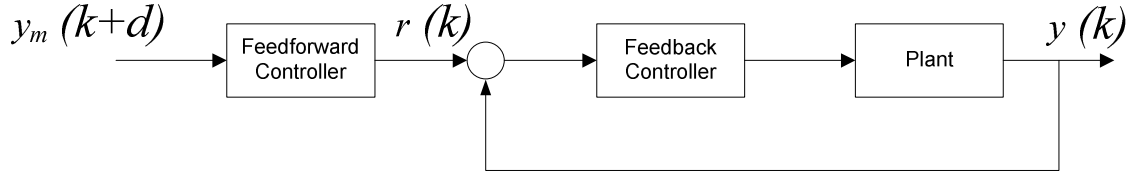
$$y(k) = G(z^{-1})r(k) \quad (2.18)$$

And the reference input from the feedforward tracking controller is,

$$r(k) = \frac{A_c(z^{-1})}{B_c(z^{-1})} y_d(k + d) \quad (2.19)$$

where  $y_d(k + d)$  is the desired output  $d$ -step ahead and its purpose is to compensate for the  $d$ -step delay in the closed-loop transfer function. From Eq. (2.18) and Eq. (2.19), it can be seen that the controller will exhibit perfect tracking where  $y(k) = y_d(k)$ , which assumes that the given initial conditions are zeros. The main disadvantage with the PTC approach is that it requires a minimum-phase system where the poles and zeroes are stable. When there is the presence of uncancellable zeroes in the system, there is tendency for the system output to be unbounded or oscillatory. These uncancellable zeros may exist outside of the unit circle or very close to it. Regardless, their existence will cause the overall response of the system to be unstable or oscillatory in nature (Astrom, Hagander, & Sternby, Zeroes of Sampled Systems, 1984). Furthermore, even if there are no unstable zeroes in the system, they may still be generated when a continuous time system is transformed into a discrete time system for purposes of digital implementation (Suh, Kang, Chung, & Stroud, 2008).

To deal with the uncancellable zeros in the PTC approach, Tomizuka (Tomizuka, 1987) proposed Zero Phase Error Tracking Controller (ZPETC) which is an approach based on pole/zero cancellation and phase cancellation. The general concept of ZPETC is that the existing feedback controller is utilized to provide the overall controller with the ability to regulate the process while the feedforward controller is used to provide the tracking control to the system. An example of this setup is shown in Figure 2-7.



**Figure 2-7: Zero phase error tracking system (Tomizuka, 1987)**

With ZPETC, the output and input of the transfer function in PTC is modified to,

$$\frac{y(k)}{r(k)} = \frac{z^{-d} B_c^-(z^{-1}) B_c^+(z^{-1})}{A_c(z^{-1})} \quad (2.20)$$

Here, the uncancellable zeroes present in  $B_c(z^{-1})$  of the PTC approach are factorized out. That is  $B^+(z^{-1})$  contains all the acceptable zeroes and  $B^-(z^{-1})$  contain all the unacceptable zeroes of the closed-loop system. This implies that the acceptable zeroes are within the unit circle in the  $z$ -plane while the unacceptable zeroes are outside or very close to the unit circle. To deal with the uncancellable zeros, Tomizuka proposed the following feedforward controller,

$$\frac{r(k)}{y_m(k+d)} = \frac{A_c(z^{-1}) B^-(z)}{B^+(z^{-1}) [B^-(1)]^2} \quad (2.21)$$

And the overall transfer of the system is,

$$\frac{y(k)}{y_m(k+d)} = \frac{B^-(z) B^-(z^{-1})}{[B^-(1)]^2} \quad (2.22)$$

The transfer function above expressed in the frequency domain as,

$$\frac{y(k)}{y_m(k+d)} = \frac{B^-(e^{-j\omega}) B^-(e^{j\omega})}{B^-(1) B^-(1)} \quad (2.23)$$

From the above equation, it is assumed that  $B^-(e^{-j\omega}) = Re + jIm$  where  $Re$  and  $Im$  are the real and imaginary components respectively. Therefore, Eq. (2.21) can be rewritten as,

$$\frac{y(k)}{y_m(k+d)} = \frac{\text{Re}^2 + \text{Im}^2}{[B^-(1)]^2} = \left\| \frac{B^-(e^{j\omega})}{B^-(1)} \right\|^2 \quad (2.24)$$

From the above frequency transfer function equation (Eq. (2.24)), it is evident that the phase angle of the transfer function is zero, thus it will yield zero tracking error. Experiments have verified that the frequency response of this transfer function has a zero phase shift for all frequencies and a unity gain at zero frequency. Experiments with ZPETC controllers were able to track highly complex contours with good tracking error and smooth velocity profiles. However, performance degradation was observed for systems with large disturbances. Since ZPETC is based on pole/zero cancellation and phase cancellation, its performance is highly sensitive to modeling errors and system parametric variations particularly for system with large disturbances.

To account for the unknown variations in modeling and system parameters, an adaptive ZPETC was introduced by Tsao and Tomizuka (Tsao & Tomizuka, 1987). This method introduces a parameter adaptation algorithm (PAA) which is based on a normalized least squares method to adjust the unknown dynamic parameters in the feedforward controller. The output and input of the adaptive ZPETC is given as follows,

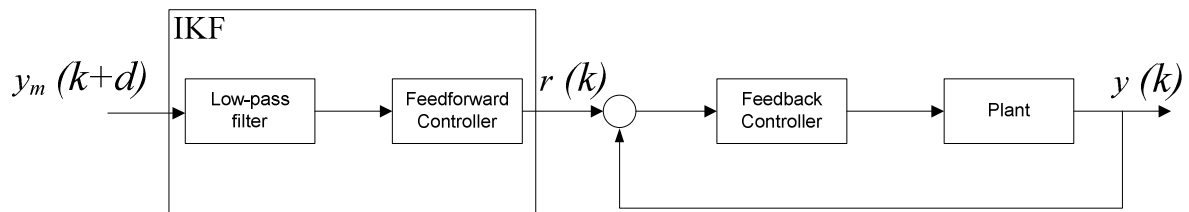
$$\frac{y(k)}{r(k)} = \frac{z^{-d}B(z^{-1})B_0^-(z^{-1})B_0^+(z^{-1})}{A(z^{-1})A_0(z^{-1})} \quad (2.25)$$

Both the numerator and denominator are divided into known and unknown parts (the known parts have a zero subscript). This separation is normally applied since the feedback controller dynamics are generally known while the dynamics of the plant are only partially known. The advantage of this adaptation algorithm is the feedback controller loop is unperturbed since the additional feedforward controller is made as a module which can be switched on and off. With the feedback controller is untouched, the system can remain relatively

stable. Simulation results with the adaptive ZPETC controller indicate improved performance over the a non-adaptive ZPETC controller.

The original ZPETC controller was further modified by Haack and Tomizuka (Haack & Tomizuka, 1991) who proposed adding zeros to the zero phase error tracking controller. This had the effect of reducing the tracking error generated from the feedback loop that contains uncancellable zeroes. Experiments indicate improvements in tracking performance for low and mid-range frequency trajectories. However, the performance did not improve in the high frequency range.

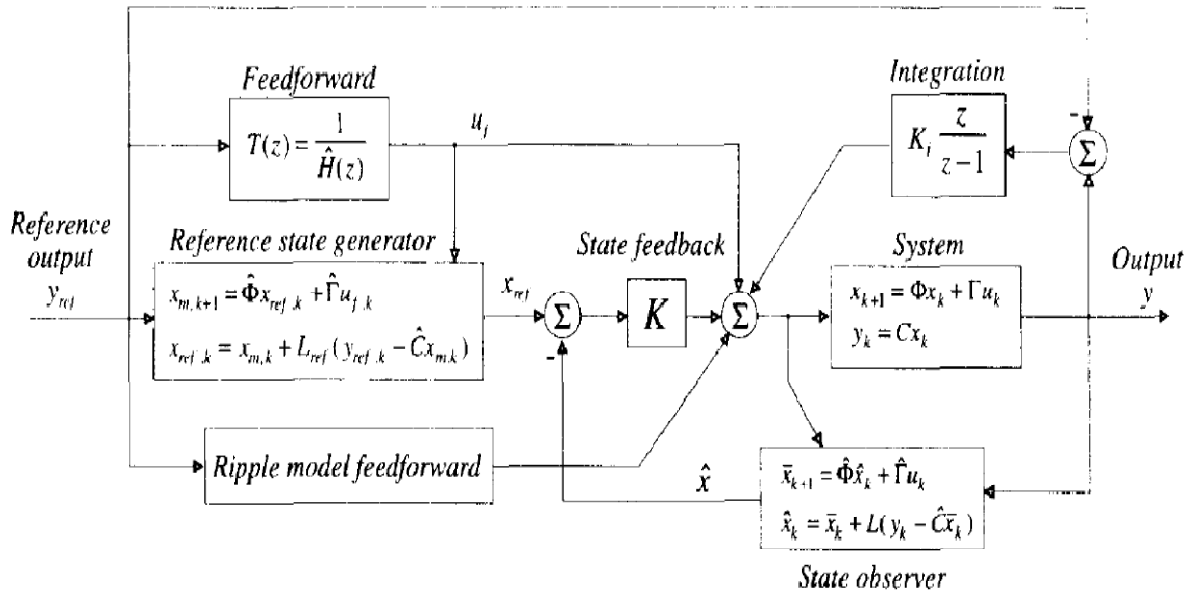
Another method that utilizes the feedforward controller principle is the Inverse Compensation Filter (IKF) developed by Weck and Ye (Weck & Ye, 1990). This method introduces a low pass filter to the feedforward controller to filter out the high frequency signals. The structure of this setup is shown in Figure 2-8. IKF was found to be effective at filtering out discontinuities in the signal. Therefore, the controller can track, with a greater precision, dynamically high-frequency paths.



**Figure 2-8: A feedforward and feedback control loop with IKF (Weck & Ye, 1990)**

Another control strategy that utilizes the feedforward principle is Van Den Braembussche et al. (Van Den Braembussche, Swevers, Van Brussel, & Vanherck, 1996) who proposed a control scheme that consists of a state feedback, feedforward, and motor ripple compensation. As with the previous feedforward methods, the feedforward loop is used to archive zero tracking

error, but with an added inverse model pre-filtering similar to the IKF strategy. This model also compensates for the cogging or motor rippling effect caused by magnetic disturbances of electric motors. These compensation models are derived through experimentation and are added as an addition feedforward loop shown Figure 2-9.



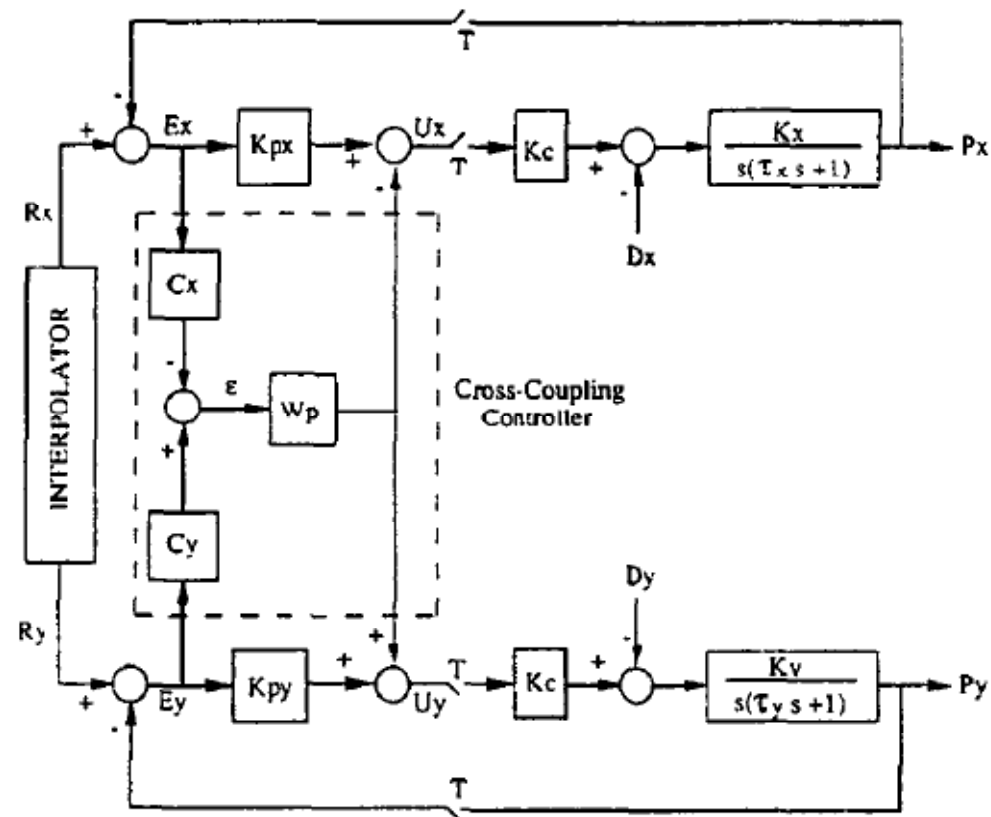
**Figure 2-9: The control scheme proposed by Van Den Braembussche (Van Den Braembussche, Swevers, Van Brussel, & Vanherck, 1996)**

Most of the control strategies presented as well as the one found in literature are based on the principle of feedforward control. Some of others not mention are predictive control and optimal control. These methods along with the ones mentioned above, have their own advantages and disadvantages as well as particular application (low versus high speed machining and low versus high variation is system dynamic parameters). Although feedforward controllers provided relative improvement in tracking performance, one major disadvantage of feedforward controllers, as mentioned, is that it requires precise knowledge of the dynamic behaviour of the

system. Even with an adaptive control scheme, the dynamics is difficult to model with sufficient accuracy as the behaviour of the real system is predominantly nonlinear and varying with time. There are also sensitivity issues that can arise from adaptive control with large system disturbances.

### 2.3      **CONTOURING ERROR CONTROL**

One of the most important contributions to CNC contouring performance in terms of control implementation advancement was made by Koren (Koren Y. , Cross-coupled biaxial computer control for manufacturing systems, 1980) who proposed cross-coupled control (CCC) for biaxial systems. The concept of CCC was designed for a sampled-data type CNC control system where each axis in a multi-axis system is controlled independently. Each axis will consist of its own closed-loop control algorithm strictly capable of detecting and correcting for its own disturbances. Disturbances from the other axes go undetected and will inevitably result in inconsistent or poor overall synchronization of all axes. With CCC, the errors that occur in one axis will affect the control loops of both axes. This means that the controller is capable of monitoring the motions of all axes relative to each other and cross-couples the error in the controller dynamics. The structure of CCC is shown in Figure 2-10. Therefore, the main objective of CCC is to eliminate the overall contour error rather than focusing on the individual axial tracking error. This is achieved by constructing a model of the contour error  $\epsilon$  in real time and then using this information to implement a control law to minimize the contouring error. The contour error model  $\epsilon$  is derived from the tracking errors  $e_x$  and  $e_y$  in a biaxial system. The result is then multiplied by a proportional gain ( $W_p$ ) and fed back into the system. In theory, it would be possible to achieve zero contour error even with the existence of large axial tracking errors from all the axes.

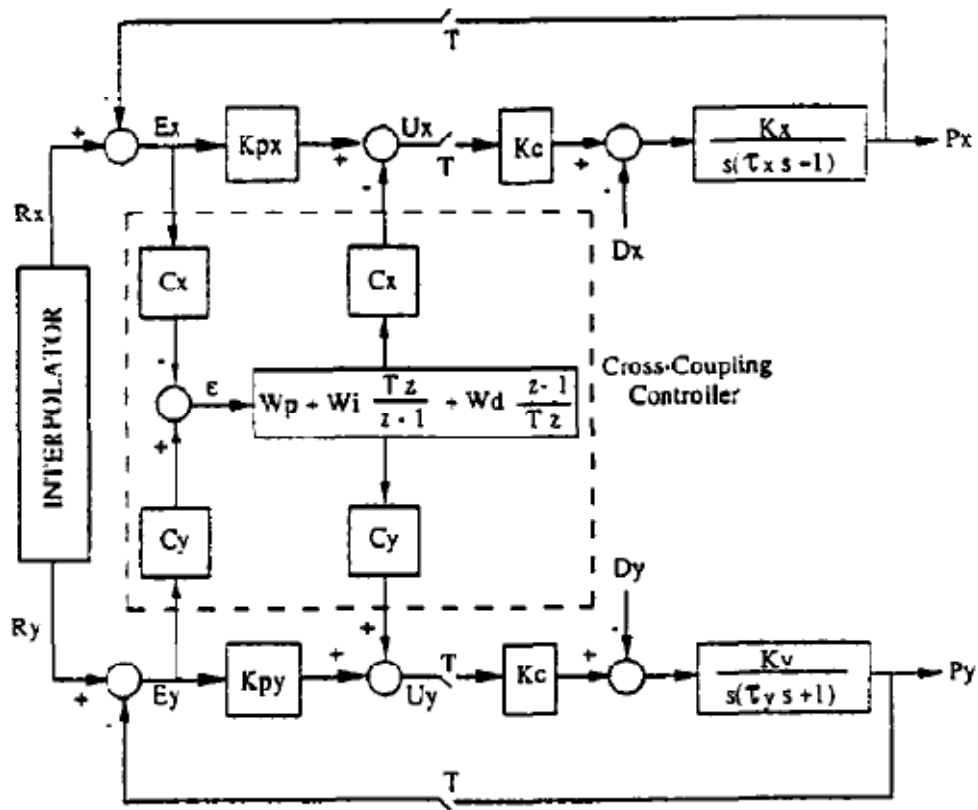


**Figure 2-10: Biaxial cross-coupled control structure (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991)**

Experiments conducted by Koren (Koren Y. , Cross-coupled biaxial computer control for manufacturing systems, 1980) with the proposed biaxial CCC system indicate contouring performance improvement over conventional feedback controllers (i.e. P-controller and PID-controller) without CCC. This performance improvement was not observed for the individual axes of the system (i.e. tracking performance improvement for one axis), but rather for the system as a whole. One advantage of the CCC strategy is the ease of implementation where it requires no addition hardware for an existing CNC-based system. For NC-based systems, additional hardware would be required. A significant disadvantage to CCC presented above is its low effectiveness in dealing with non-linear contours (i.e. circles and parabolas) due to the

difficulty of analytically determining the contour error in real time. Even for linear contours, where the contour error is relatively simple to determine, there is a tendency for the contour error to oscillate as the steady-state tracking error goes to zero.

To improve upon the original concept of CCC proposed by Koren, Koren and Lo (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991) introduced a revised version of CCC called variable-gain cross-coupled control. In variable-gain CCC, the cross-coupled control gains are adjusted in real-time depending on the shape of the contour. This implies that CCC can be utilized for nonlinear contours such as circles and parabolas. Figure 2-11 below shows the addition of variable gains  $C_x$  and  $C_y$  to the original CCC scheme.



**Figure 2-11: Variable-gain cross coupled control (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991)**

Here,  $C_x$  and  $C_y$  are defined by the instantaneous shape of the contour. For linear contours, these gains are constant and are defined as,

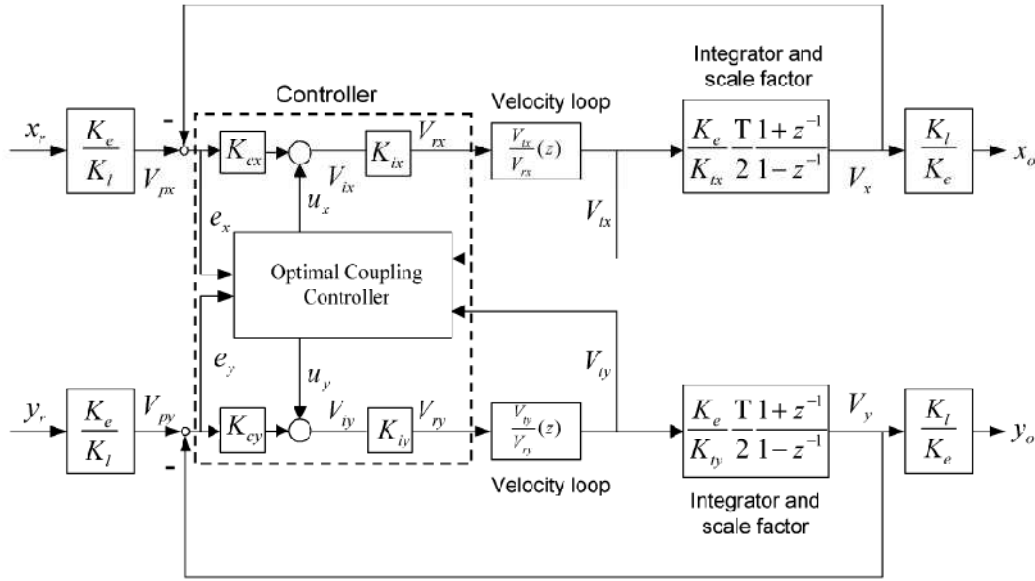
$$\begin{cases} C_x = \sin \theta \\ C_y = \cos \theta \end{cases} \quad (2.26)$$

where  $\theta$  is defined in Figure 2-2. For circular contours,  $C_x$  and  $C_y$  are functions of both  $\theta$  and  $R$ , where  $R$  is defined as the radius of the circle. The variable CCC gains are then given as,

$$\begin{cases} C_x = \sin \theta - \frac{e_x}{2R} \\ C_y = \cos \theta + \frac{e_y}{2R} \end{cases} \quad (2.27)$$

where  $e_x$  and  $e_y$  are the tracking error in the  $x$ -axis and  $y$ -axis respectively. For other nonlinear contours, the values  $C_x$  and  $C_y$  are estimated by approximating the local portion of the contour as circular. Therefore, the gains are still functions of  $\theta$  and  $R$ , where  $R$  now is the instantaneous radius of the curvature. Simulations of linear, circular, and parabolic type contours with variable CCC gains by Koren and Lo (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991) indicate significant reduction in contouring errors by as much as a 10:1 ratio with the lowest ratio achieved of 3:1.

Different control structures have also been proposed base on the Koren and Lo's variable-gain cross-coupling. Kulkarni and Srinivasan (Srinivasan & Kulkarni, 1990) introduced optimal coupled control with the structure shown in Figure 2-12. Results from experiments found that the proposed coupled optimal controller was effective at reducing contouring error for linear, corners, and circular contours for low machine feed rates ranging from 2.25 to 5.63m/min for linear and 7.2m/min for circular contours with a radius of 30mm. Furthermore, it was found that in order to obtain comparable contouring errors in the uncoupled controller, significantly higher control gains had to be utilized.



**Figure 2-12: Optimal coupled control(Srinivasan & Kulkarni, 1990)**

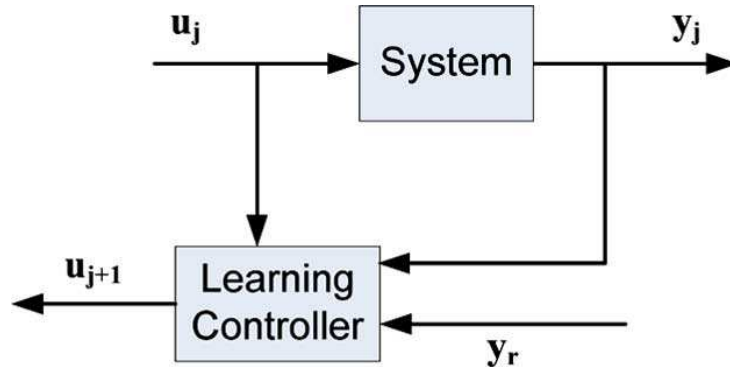
Although highly successful at reducing the contouring error, the principle of CCC could only be applied to biaxial systems since the contouring error can be determined analytically. For systems with three or more axes, calculating the contouring error in real time based on all axial tracking errors becomes very difficult. The best solution to is provide an estimation of the contour error in real-time with an estimation method outlined a later section.

## 2.4 ADVANCED CONTROL METHODS

Sections 2.2 and 2.3 discussed some of the developed control techniques relevant to current CNC control systems. This section will review some of the advanced techniques, namely iterative learning control (ILC), repetitive control (RC), and event-driven control (EDC).

The concept of ILC is based on the fact that when a machine performs a task that is repeated many time, such is the case for many CNC applications, it can analyze and store the

errors that were made and make appropriate corrections. Through this method, the errors in the system outputs can be made progressively smaller with each successive iteration. This forms a category of learning-type control strategies, which also includes adaptive and repetitive control. ILC is different from these control strategies since it only modifies the control input to a system. With adaptive control, the controller itself is modified (Bristow, Tharayil, & Alleyne, 2006). One important consequence of ILC and other learning-type control strategies is that it is only capable of correcting for repetitive errors. The general structure of the ILC can be seen in Figure 2-13.



**Figure 2-13: Iterative learning control structure (Barton & Alleyne, 2008)**

One of the first publications to introduce ILC to CNC control was proposed by Kim et al. (Kim & S., 1996). This involved incorporating a learning function into a conventional PID controller. The input of the controller is given as,

$$u_{k+1}(i) = u_k(i) + K_P e_k(i+1) + K_I \sum_{n=1}^{i+1} e_k(n) + K_D [e_k(i+1) - e_k(i)] \quad (2.28)$$

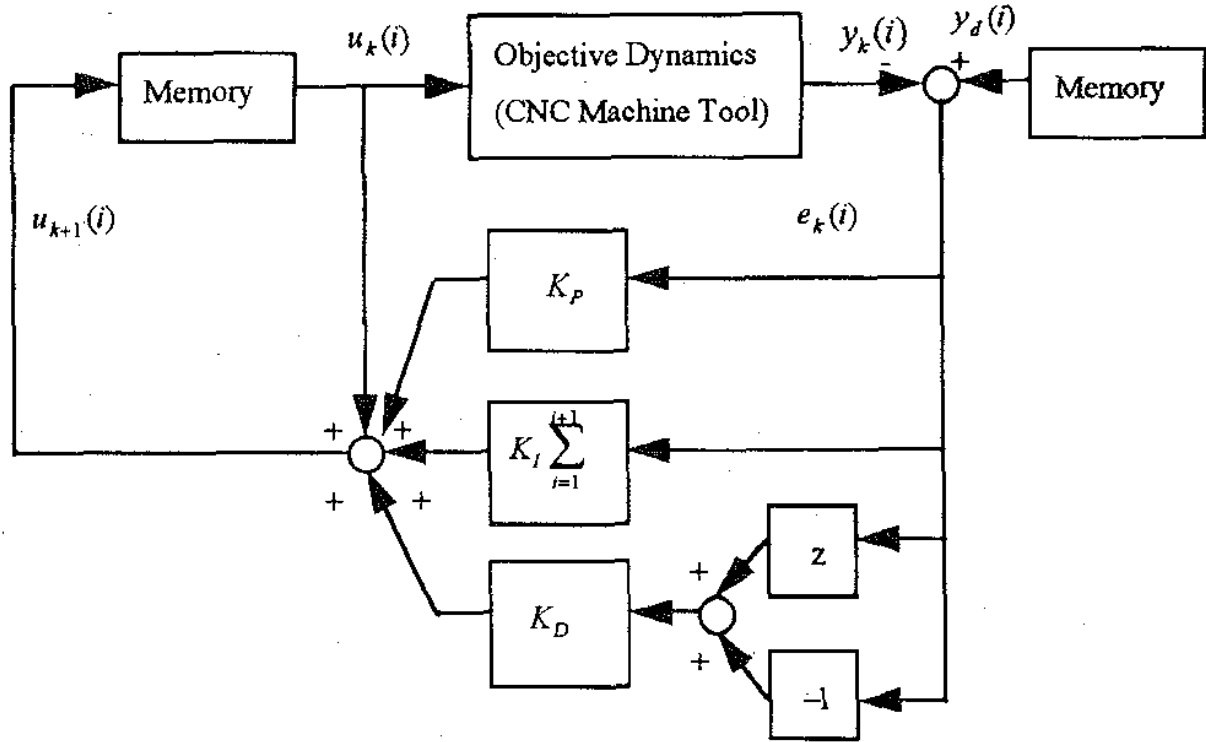
where  $k$  is the iteration step and  $i$  is the discrete time.  $e_k(i)$  is the system output error. The overall system is expressed as,

$$\begin{aligned}
x_k(i+1) &= A(i)x_k(i) + B(i)u_k(i) = D(i)w_k(i) \\
x_k(0) &= x_o \\
y_k(i) &= C(i)x_k(i) \\
u_k(i) &= u_{k-1}(i) + K_P e_{k-1}(i+1) + K_I \sum_{n=1}^{i+1} e_{k-1}(n) \\
&\quad + K_D [e_{k-1}(i+1) - e_{k-1}(i)] \\
e_{k-1}(i) &= y_d(i) - y_{k-1}(i)
\end{aligned} \tag{2.29}$$

The structure of this system is displayed in Figure 2-14. Experiment with this controller yielded improved performance after only 4 iterations with a 58% increase in the machining accuracy for circular type contours.

Barton and Alleyne (Barton & Alleyne, 2008) proposed a control strategy that can improve both tracking and contouring errors by incorporating both ILC and CCC called cross-coupled iterative learning control (CCILC). CCILC uses a more common ILC learning algorithm as shown below,

$$u_{k+1}(i) = Q(q^{-1})[u_k(i) + L(q)e_k(i+1)] \tag{2.30}$$



**Figure 2-14: Structure of a learning-type PID controller (Kim & S., 1996)**

where  $j$  is the iteration step,  $e_j$  is the tracking error which is calculated one time step ahead to compensate for the delay in the plant dynamics.  $q$  is the forward time-shift operator where for a given system,  $qx(i) = x(i + 1)$ .  $Q$  and  $L$  are functions defined as the  $Q$ -filter and learning function respectively. The  $Q$  function is designed to give the system stability when there are uncertainties in the plant dynamics. The  $L$  function is designed in such a way to achieve good tracking performances by modifying the previous controller input  $u_k(i)$ . The control structure of ILC is displayed in Figure 2-15 and Figure 2-16. At the end of each iteration, the error is filtered through the learning function  $L$  and then added to the control input. This is then filtered through the  $Q$  function to form a new controller input for the next iteration.

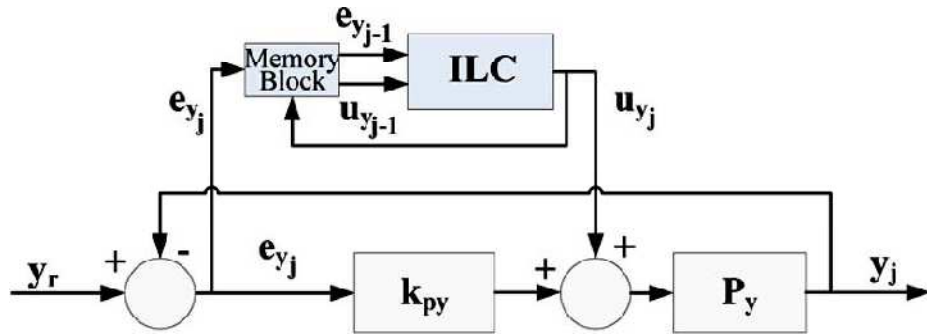


Figure 2-15: ILC controller scheme with a feedback loop (Barton & Alleyne, 2008)

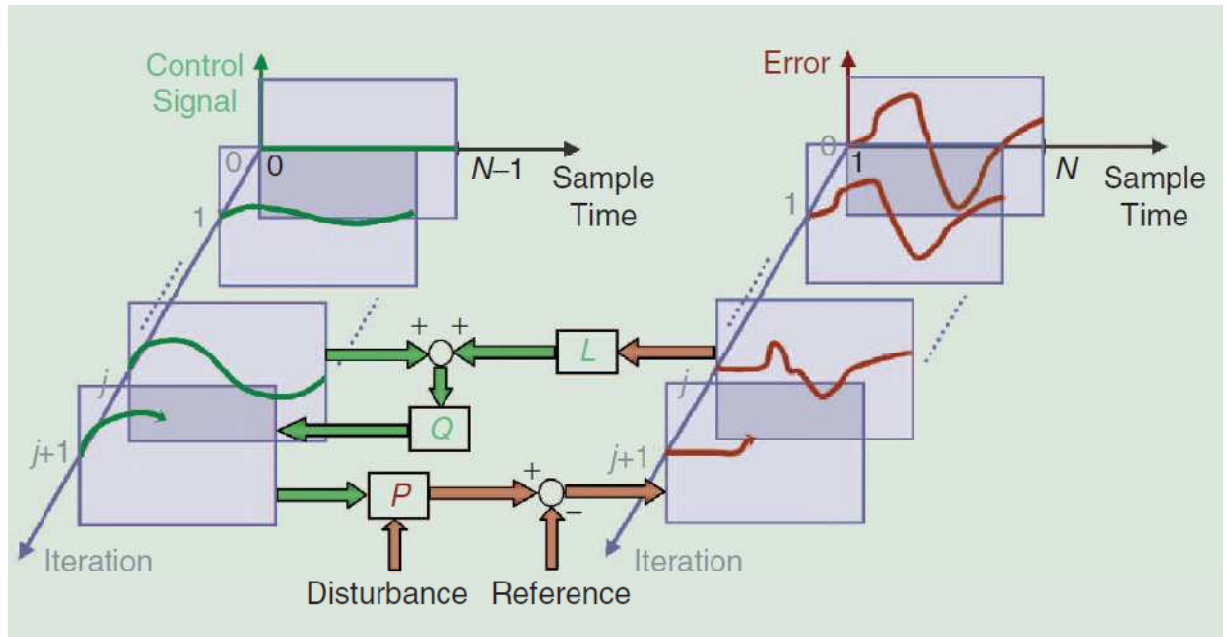
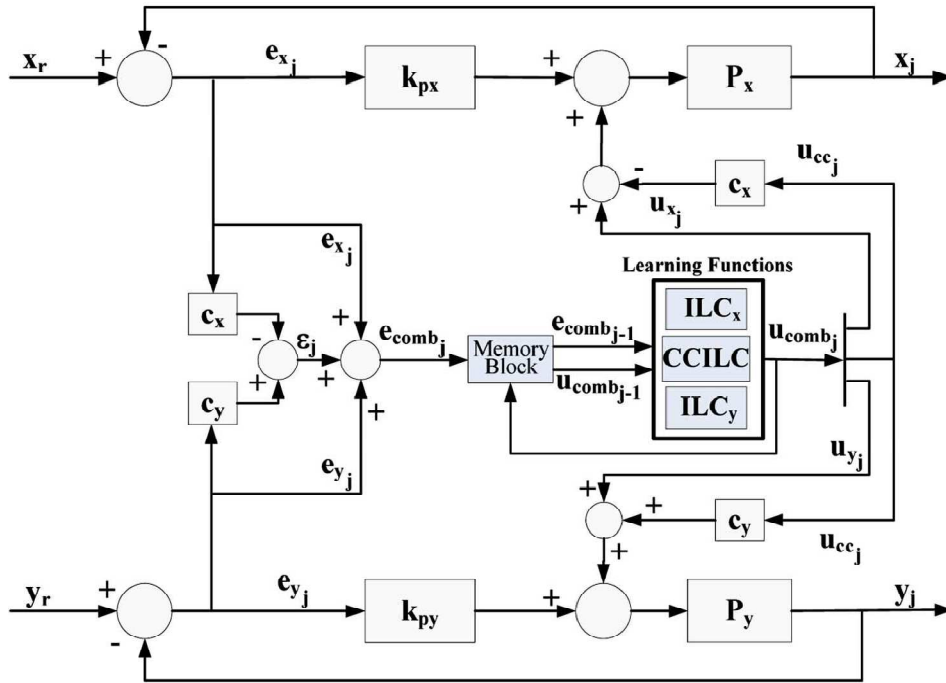


Figure 2-16: An overall structure of and ILC system (Bristow, Tharayil, & Alleyne, 2006)

With the addition of the CCC, the CCILC control law is,

$$u_{cc_{k+1}}(i) = Q(q^{-1})[u_{cc_k}(i) + L(q)\epsilon_j(i+1)] \quad (2.31)$$

In the CCILC control law, the contouring error  $\epsilon$  is use in place of the tracking error with the overall control diagram display in Figure 2-17. Here, the contouring error is calculated using the same method described in (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991). Both experimentation and simulation of CCILC yielded improvement over conventional feedback CCC controllers without the learning controller. Results indicate a reduction in axial tracking error range from 35% to 76%.



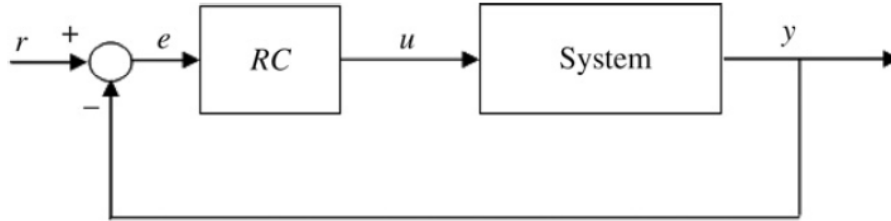
**Figure 2-17: Iterative learning control with cross-coupled control(Barton & Alleyne, 2008)**

In general, ILC is a highly active area of research spanning the past three decades with developed theories in stability, robustness, and system limitations. It will only continue develop, particularly in the area of micro and nano-manufacturing, where the small scale is considered. Further review on ILC can be found in (Bristow, Tharayil, & Alleyne, 2006) and (Longman, 2000).

Another learning-type controller is called repetitive control (RC), which was first introduced by (Inoue, Nakano, Kubo, Matsumoto, & Baba, 1981). The RC update law, which is every similar to ILC, is given as,

$$u_{k+1}(i) = u_k(i) + \lambda e_k(i + 1) \quad (2.32)$$

where  $k$  is the iteration step and  $i$  is the discrete time.  $e_k$  is the tracking error and  $\lambda$  is called the learning gain satisfying  $0 < \lambda < 1$ . From this RC update law, it can be easily seen to resemble the ILC learning algorithm from Eq. (2.30) with one exception that the RC law lacks the  $Q$  filter functions found in ILC. Again, the addition of the  $Q$  function is designed to give the system stability when there are uncertainties in the plant dynamics. One of the major differences between ILC and RC is that in ILC, the initial conditions are the same for every iteration where as in RC, the initial conditions are set to the final values from the previous iterations. This is the case since RC was intended to be used on continuous operation (Bristow, Tharayil, & Alleyne, 2006). The general block diagram of RC is shown in Figure 2-18.



**Figure 2-18: Repetitive control diagram (Tan, Chua, Zhao, Yang, & M.T, 2009)**

Another advanced control method, called event-driven control or periodic control. This control method differs from traditional control methods in that it requires the occurrence of an event to trigger a new control action. Traditional control methods have fixed time-triggered sampling, which means that although nothing is happening in the system, the controller will still

perform a new control action at the prescribed time interval. This can lead to unnecessary utilization of system resources (Arzen, 1999).

Experiments conducted with a simple event-driven PID controller by Årzén (Arzen, 1999) indicate significant reduction in CPU utilization by the event-driven controllers with only minor control performance degradation.

## 2.5 REMARKS

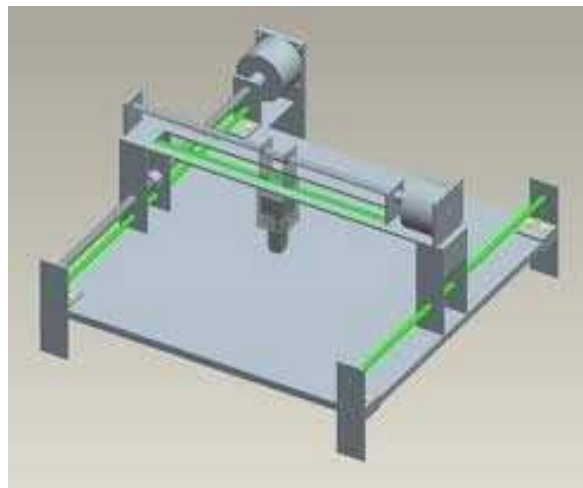
Most of the above control strategies typically proposed a method or algorithm to improve one aspect of the CNC system for the improvement of overall performance of the system. Cheng and Chin (Cheng & Chin, 2003) proposed merging all these aspects in an effort to analyze and reduce both tracking and contouring error. This research suggests that by only considering only one aspect and inevitably ignoring others, any improvements made to the system can be nullified as a result. To make a significant improvement, the system as a whole must be considered. With the increase in computing power and general knowledge of CNC systems, it is conceivable that future research will be geared towards this direction of thinking.

Existing control systems are constructed in time domain with limitations as discussed above. To advance the control system, in the next chapter, a new control law based in position domain will be proposed as an alternative to the traditional control systems. The main motivation of this research is to take advantage CCC and event-driven control systems to form a new control method that can improve both the tracking and contour tracking performance of the system.

## Chapter 3 : POSITION DOMAIN CONTROL SYSTEM DESIGN

### 3.1 DYNAMICS MODEL

PDC is a novel control scheme whereby a multi-axis system is viewed as a master-slave system. The motion of the master axis is equidistantly sampled and used as an independent reference. The motion of each of the slave axis is defined as a function of the reference motion based on the contouring requirements of the system. To formulate this function, the dynamic model of the multi-axis system is developed in position domain through a one-to-one mapping from time domain. One major advantage to the PDC strategy is that there is no tracking error in the reference motion, thus the reference motion does not affect or contribute to the overall contouring error. Therefore, in order to get an accurate contouring performance, a high precision measurement is required from the reference motion in order for a PDC system to be effectively utilized.



**Figure 3-1: Schematic of a CNC machine**

From the CNC system represented in Figure 3-1, a simplified dynamic model of this system in time domain is given as,

$$\begin{cases} m_x \ddot{q}_x + c_x \dot{q}_x + k_x q_x = T_x(t) \\ m_y \ddot{q}_y + c_y \dot{q}_y + k_y q_y = T_y(t) \\ m_z \ddot{q}_z + c_z \dot{q}_z + k_z q_z = T_z(t) \end{cases} \quad (3.1)$$

where  $c_i$  and  $k_i$  are the damping and stiffness coefficient of the system respectively.  $\ddot{q}_i$ ,  $\dot{q}_i$ , and  $q_i$  are the acceleration, velocity, and position of the  $i^{th}$  axis, and  $T_i$  is its control input.

The state space representation of the system is,

$$\begin{cases} \dot{X} = AX + BU \\ y = CX \end{cases} \quad (3.2)$$

$$\text{where, } X = \begin{bmatrix} q_x \\ \dot{q}_x \\ q_y \\ \dot{q}_y \\ q_z \\ \dot{q}_z \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{k_x}{m_x} & -\frac{c_x}{m_x} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{k_y}{m_y} & -\frac{c_y}{m_y} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\frac{k_z}{m_z} & -\frac{c_z}{m_z} \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{m_x} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{m_y} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m_z} \end{bmatrix},$$

$$U = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}, \text{ and } C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

Other dynamic factors such as backlash and friction are excluded from the model to simplify simulation and the comparative analysis between different control laws.

### 3.2 TIME DOMAIN CONTROL

The dynamic model in Eq. (3.1) assumes that dynamics of each axis is decoupled. Therefore, a conventional PID controller in time domain can be applied to control each axis. The controller is given as,

$$T_i(t) = K_P^i e_i(t) + K_I^i \int_0^t e_i(\tau) d\tau + K_D^i \dot{e}_i(t) \quad (3.3)$$

where  $K_P^i$ ,  $K_I^i$ , and  $K_D^i$  are the PID controller gains.  $e_i$  and  $\dot{e}_i$  are the error and derivative of the error.

### 3.3 TIME DOMAIN CONTROL WITH CROSS-COUPLED CONTROL

The CCC control laws used in this research is derived from (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991). For linear contour tracking, the contour error and its derivative are derived from the tracking error of the  $j$  and  $k$  axes as,

$$\begin{aligned} e_c &= -C_j e_j + C_k e_k \\ \dot{e}_c &= -C_j \dot{e}_j + C_k \dot{e}_k \end{aligned} \quad (3.4)$$

where  $C_j = \sin \theta$  and  $C_k = \cos \theta$  refers to the variable-gains in the cross-coupled control. For a linear motion, both  $C_j$  and  $C_k$  are constant since  $\theta$  is constant.

The corresponding circular motion contour errors are defined as,

$$\begin{aligned} e_c &= -C_j e_j + C_k e_k \\ \dot{e}_c &= -C_j \dot{e}_j + C_k \dot{e}_k - \dot{C}_j e_j + \dot{C}_k e_k \end{aligned} \quad (3.5)$$

where  $C_j = \sin \theta - \frac{e_j}{2R}$ ,  $C_k = \cos \theta + \frac{e_k}{2R}$ ,  $\dot{C}_j = \dot{\theta} \cos \theta - \frac{\dot{e}_j}{2R}$ , and  $\dot{C}_k = -\dot{\theta} \sin \theta + \frac{\dot{e}_k}{2R}$ . From these relationships, it is possible to develop the PID control law that includes cross-coupling control. The TDC controller with CCC for the  $j$  and  $k$  axes are given as,

$$T_j(t) = K_P^j e_j(t) + K_I^j \int_0^t e_j(\tau) d\tau + K_D^j \dot{e}_j(t) - C_j (K_P^j e_c + K_D^j \dot{e}_c) \quad (3.6)$$

And

$$T_k(t) = K_P^k e_k(t) + K_I^k \int_0^t e_k(\tau) d\tau + K_D^k \dot{e}_k(t) + C_k(K_P^k e_c + K_D^k \dot{e}_c) \quad (3.7)$$

### 3.4 POSITION DOMAIN CONTROL

In order to utilize a PDC controller, the dynamics of the CNC system presented in Eq. (3.1) must be converted from time domain into position domain. In TDC, the position of each axis is time dependent. This relationship given in time domain is,

$$q_j = f_j(t) \quad (3.8)$$

The objective of PDC is to replace the time-dependency of the system with a dependency on one of its axis referred to as the reference or master axis. The motions of other axes, called slave axes, can be represented as functions of the motion of the master axis. Assuming that  $x$ -axis in Eq. (3.1) is the reference axis, then the position for the  $j^{th}$  axis in position domain is given as,

$$q_j = g_j(q_x) \quad (3.9)$$

To formulate a control law based on the position domain, it is necessary to define the relative velocity and relative acceleration of the  $j^{th}$  axis in the position domain from Eq. (3.9) as,

$$q_j' = \frac{dq_j}{dq_x} \quad (3.10)$$

And

$$q_j'' = \frac{d^2 q_j}{dq_x^2} \quad (3.11)$$

The  $i^{th}$  axis velocity and acceleration can be represented as functions of the relative velocity and relative acceleration as follows,

$$\dot{q}_j = \frac{dq_j}{dt} = \frac{dq_j}{dq_x} \frac{dq_x}{dt} = q_j' \dot{q}_x \quad (3.12)$$

And

$$\ddot{q}_j = \frac{d\dot{q}_j}{dt} = \frac{d(q_j' \dot{q}_x)}{dt} = \dot{q}_x \frac{dq_j'}{dt} + \frac{d\dot{q}_x}{dt} q_j' \quad (3.13)$$

Substituting Eq. (3.12) and Eq. (3.13) into Eq. (3.1) yields,

$$m_j \dot{q}_x^2 q_j'' + m_j \ddot{q}_x q_j' + c_j \dot{q}_x q_j' + k_j q_j = T_j(q_x) \quad (3.14)$$

Therefore, the dynamics of  $i^{th}$  axis in position domain can be expressed as,

$$m_j \dot{q}_x^2 q_j'' + (m_j \ddot{q}_x + c_j \dot{q}_x) q_j' + k_j q_j = T_j(q_x) \quad (3.15)$$

**Remark 1:** Eq. (3.15) above represents the dynamic relationship between two axes ( $x$  and  $i$ ) in position domain by transforming the dynamic model from time domain to position domain. From this, it can be concluded that to obtain accurate contour performance, a high precision measurement is required in the master ( $x$ -axis) motion direction.

**Remark 2:** Since the  $x$ -axis position is used as a reference, a high tracking accuracy of the  $x$ -axis (master motion) is not a necessary requirement in the position domain control. This implies that a low cost driving system with low accuracy can be applied for the actuation of the  $x$ -axis.

**Remark 3:** By comparing Eq. (3.1) and Eq. (3.15), it can be concluded that by the transformation from time domain to position domain, the dynamic model of the system in the position domain is nonlinear in compared to the original linear system in time domain.

From the transformation to the position domain dynamics model, the proposed control law in the position domain is given as

$$T_j(x) = K_P^j e_j(x) + K_I^j \int_0^x e_j(\tau) d\tau + K_D^j e_j'(x) \quad (3.16)$$

With the errors and the derivative of the error are given as

$$\begin{cases} e_j(x) = q_{j_d}(x) - q_j(x) \\ e_j'(x) = q_{j_d}'(x) - q_j'(x) \end{cases} \quad (3.17)$$

**Remark 4:** The PDC law of Eq. (3.16) is similar to the TDC law of Eq. (3.3) with some fundamental differences. The first is the domain that each uses is different; and secondly, the derivative gains have different physical meanings due to the differences between the derivative of the errors ( $\dot{e}_i(t)$  and  $e_i'(x)$ ) in TDC and PDC respectively.

**Remark 5:** The tracking error from Eq. (3.17) forms the contour error in the position domain where the  $x$ -axis is the reference axis with zero bias. That is  $e_c(x) = e(x)$  for a planar contour.

Substituting Eq. (3.17) into Eq. (3.16) gives the control law in position domain where the  $x$ -axis is the reference axis. The PDC control law is,

$$\begin{aligned} T_i(x) = & K_P^i (q_{i_d}(x) - q_i(x)) + K_I^i \Sigma (q_{i_d}(x) - q_i(x)) \\ & + K_D^i (q_{i_d}'(x) - q_i'(x)) \end{aligned} \quad (3.18)$$

Substituting in the definition of the relative velocity and relative acceleration define in Eq. (3.10) and Eq. (3.11) into Eq. (3.18) we have the following,

$$T_i = K_P^i \left( q_{i_d}(x) - q_i(x) \right) + K_I^i \sum \left( q_{i_d}(x) - q_i(x) \right) + \frac{K_D^i}{\dot{q}_x} \left( \dot{q}_{i_d}(x) - \dot{q}_i(x) \right) \quad (3.19)$$

which represents the PDC controller. The stability of this controller is demonstrated in the proceeding sections.

**Remark 6:** It is clearly shown that the position domain linear PID controller is equivalent to a nonlinear PID controller in the time domain when the speed of the x-axis is not constant. Dynamic model in the position domain (Eq. (3.15)) is equivalent to the dynamic model in the time domain (Eq. (3.1)). Therefore, it can be concluded the proposed PID control in the position domain has the same stability property as the PID control developed in the time domain.

**Remark 7:** If the motion of x-axis is constant, the dynamic model of Eq. (3.15) in the position domain will become a linear differential equation, and the PID control law in the position domain can be transferred to a linear PID control law in the time domain as shown in Eq. (3.18).

**Remark 8:** The proposed position domain PID control law in Eq. (3.19) can be viewed as a varying sampling rate PID control in the time domain (Schinkel, Chen, & Rantzer, 2002). Also, the position domain PID control law can be viewed as a nonlinear PID control in the time domain (Ouyang, Zhang, & Wu, 2002) when the speed of X-axis is not constant.

## 3.5 STABILITY ANALYSIS

### 3.5.1 PREPARATION AND LEMMA

Consider a dynamic system described in the position domain by,

$$y'(x) = f(x, y(x)) \quad (3.20)$$

where  $x \in R$  is the “position” of the master motion or the independent variable, and  $y(x) \in R^*$  is the state.

**Lemma 1:** Let  $D \subset R^*$  be a domain that contains the origin and  $V: [0, \infty) \times D \rightarrow R$  be continuously differentiable function such that,

$$\gamma_1(\|y\|) \leq V(y) \leq \gamma_2(\|y\|) \quad (3.21)$$

And,

$$V'(y) \leq -W(y), \forall \|y\| \geq \mu > 0 \quad (3.22)$$

with  $\forall x \geq 0$  and  $\forall y \in D$ , where  $\gamma_1$  and  $\gamma_2$  are class  $K$  functions and  $W(y)$  is a continuous positive definite function. Take  $r > 0$  such that  $B \subset D$  and suppose that,

$$\mu < \gamma_2^{-1}(\gamma_1(r)) \quad (3.23)$$

Then, there exists a class  $KL$  function  $\phi$  and for every initial state  $y(x_0)$ , satisfying  $\|y(x_0)\| \leq \gamma_2^{-1}(\gamma_1(r))$ , there is  $X \geq 0$  such that the solution of the dynamic equation satisfies,

$$\|y(x)\| \leq \phi(\|y(x_0)\|, x - x_0), \forall x_0 \leq x \leq x_0 + X \quad (3.24)$$

$$\|y(x)\| \leq \gamma_1^{-1}(\gamma_2(\mu)), \forall x \geq x_0 + X \quad (3.25)$$

Moreover, if  $D = R^n$  and  $\gamma_1$  belongs to class  $K_\infty$ , then Eq. (3.24) and Eq. (3.25) hold for any initial state  $y(x_0)$ , with no restriction on how large  $\mu$  is.

**Proof:** See reference (Khalil, 2002)

This Lemma means that the dynamic system is globally uniformly exponentially convergent to a closed ball for any initial conditions, if one can find a positive definite function  $V(y)$  so that Eq. (3.21) to Eq. (3.25) are satisfied.

### 3.5.2 NOTATIONS AND THEOREM

To facilitate the discussion, we have the following assumptions:

- A1. The desired contour of  $y_d(x)$  is of the third-order continuity for  $x \in [x_{ini}, x_{fin}]$ .
- A2. The real tracking path in the  $x$ -axis  $x(t)$  is of the third-order continuity for the contour.
- A3. The disturbance  $f_y(x)$  is bounded in the full contour tracking process.

For the briefness of the stability analysis, some notations are introduced and used in the following sections.

- $\|z\| = \max_{1 \leq i \leq n} |z|$
- $\rho = m_y \dot{x}^2 y_d''(x) + (m_y \ddot{x} + c_2 \dot{x}) y_d'(x) + k_2 y_d(x) + f_y|_{max}$
- $|\rho|_{max} = |m_y \dot{x}^2 y_d''(x) + (m_y \ddot{x} + c_2 \dot{x}) y_d'(x) + k_2 y_d(x) + f_y|_{max} \leq$   
 $m_y \|\dot{x}^2\| \|y_d''(x)\| + (m_y \|\ddot{x}\| + c_2 \|\dot{x}\|) \|y_d'(x)\| + k_2 \|y_d(x)\| + \|f_y\|$
- $\rho_e = K_{py} + k_y + \frac{\alpha}{2\beta} - \frac{m_y \ddot{x} + c_y \dot{x} + K_{Dy}}{2} - \frac{K_{Iy}}{\beta}$
- $\rho_{e'} = \left(1 - \frac{\beta}{2}\right) (m_y \ddot{x} + c_y \dot{x} + K_{Dy}) - \beta m_y \dot{x}^2 + \frac{\alpha}{2}$ , where  $\alpha$  and  $\beta$  are positive constant with  $0 < \beta < 1$

According to the assumptions (A1-A3), one can prove that the parameter  $|\rho|_{max}$  is bounded. Parameter  $\rho_{e'}$  is related to  $K_{Dy}$  while  $\rho_e$  is related to all three control gains. For the developed position domain PID controller Eq. (3.19), we have the following theorem.

**Theorem:** A system represented in position domain by Eq. (3.15), where the desired contour shape satisfies assumptions A1 and A2, is controlled by the proposed position domain

PID control law in Eq. (3.19). The contour tracking error and its relative derivative are bounded, and the boundedness are given by,

$$\begin{cases} \|e_y\| \leq 2 \sqrt{\frac{1}{\rho_e^2} + \frac{1}{\beta \rho_e \rho_{e'}}} |\rho|_{max} \\ \|e'_y\| \leq 2 \sqrt{\frac{\beta}{\rho_e \rho_{e'}} + \frac{1}{\rho_{e'}^2}} |\rho|_{max} \end{cases} \quad (3.26)$$

Provided that the control gains and the positive constant parameters are selected properly such that,

$$\begin{cases} K_{Dy} > \frac{\alpha}{\beta} + m_y \|\ddot{x}\| + c_y \|\dot{x}\| \\ K_{Iy} < \alpha \beta \\ K_{Py} > \frac{K_{Iy}}{\beta} + \frac{1}{2} (m_y \|\ddot{x}\| + c_y \|\dot{x}\| + K_{Dy}) \\ \frac{\alpha}{\beta} > \frac{2\beta}{1-\beta} m_y \|\dot{x}\|^2 \end{cases} \quad (3.27)$$

It is noticed that Eq. (3.27) provides some guidelines about the choices of control gains for the developed position domain PID control.

### 3.5.3 PROOF OF STABILITY

A stability analysis for the proposed position domain PID control is conducted based on the Lyapunov function method. First define,

$$\begin{cases} \sigma_y(x) = \int_0^x e_y(s) ds \\ \sigma'_y(x) = e_y(x) \end{cases} \quad (3.28)$$

Using Eq. (3.28), the dynamics model with position domain PID controller in Eq. (3.15) can be re-described in an error function format as follows,

$$m_y \dot{x}^2 e_y''(x) + (m_y \ddot{x} + c_y \dot{x} + K_{Dy}) e_y'(x) + K_{Ly} \sigma_y(x) + (K_{Py} + k_y) e_y(x) = \rho \quad (3.29)$$

For the dynamic system described in position domain, we define the following Lyapunov functions,

$$\begin{cases} V_1(e_y(x), e_y'(x)) = \frac{1}{2} [e_y & e_y'] \begin{bmatrix} K_{Py} + k_y & \beta m_y \dot{x}^2 \\ \beta m_y \dot{x}^2 & m_y \dot{x}^2 \end{bmatrix} \begin{bmatrix} e_y \\ e_y' \end{bmatrix} \\ V_2(e_y(x), \sigma_y(x)) = \frac{1}{2} [e_y & \sigma_y] \begin{bmatrix} \alpha & K_{Ly} \\ K_{Ly} & \beta K_{Ly} \end{bmatrix} \begin{bmatrix} e_y \\ \sigma_y \end{bmatrix} \end{cases} \quad (3.30)$$

$$V(e_y(x), e_y'(x), \sigma_y(x)) = V_1(e_y(x), e_y'(x)) + V_2(e_y(x), \sigma_y(x)) \quad (3.31)$$

If the control gains are properly chosen according to Eq. (3.27), as  $0 < \beta < 1$ , the following inequality holds,

$$K_{Py} \geq \beta m_y \|\dot{x}\|^2 > \beta^2 m_y \dot{x}^2 - k_y \quad (3.32)$$

According to Eq. (3.32), we can prove that  $V_1(e_y(x), e_y'(x))$  is a positive definite function. From Eq. (3.27), it is easy to prove that  $V_2(e_y(x), \sigma_y(x))$  is also a positive definite function. Therefore, the Lyapunov function  $V(e_y(x), e_y'(x), \sigma_y(x))$  is a positive definite function.

It is easy to demonstrate that,

$$\begin{cases} ab \geq -\frac{1}{2}(a^2 + b^2) \\ ab \leq \frac{1}{2}(a^2 + b^2) \end{cases} \quad (3.33)$$

Applying Eq. (3.33) to Eq. (3.31) - Eq. (3.32), the following inequalities can be obtained,

$$\begin{aligned}
& \frac{1}{2}(K_{Py} + k_y - \beta m_y \dot{x}^2)e_y^2 + \frac{1}{2}(1 - \beta)m_y \dot{x}^2 e_y'^2 \\
& + V_2(e_y(x), \sigma_y(x)) \leq V(e_y(x), e_y'(x), \sigma_y(x)) \\
& \leq \frac{1}{2}(K_{Py} + k_y + \beta m_y \dot{x}^2)e_y^2 + \frac{1}{2}(1 + \beta)m_y \dot{x}^2 e_y'^2
\end{aligned} \tag{3.34}$$

From Eq. (3.34), we can see that the defined Lyapunov function satisfies Eq. (3.21).

In the position domain control, the reference position  $x$  of the  $x$ -axis motion is an independent variable that has a similar meaning of  $t$  in time domain.  $e_y$  and  $e_y'$  are functions of the independent variable  $x$ . Therefore, the derivative of the Lyapunov function  $V$  is related to the variable  $x$  in this stability analysis.

Rewriting Eq. (3.29), we have,

$$\begin{aligned}
m_y \dot{x}^2 e_y''(x) &= \rho - (m_y \ddot{x} + c_y \dot{x} + K_{Dy})e_y'(x) - K_{Ly}\sigma_y(x) \\
&\quad - (K_{Py} + k_y)e_y(x)
\end{aligned} \tag{3.35}$$

Differentiating Eq. (3.31) with respect to the variable  $x$  and using Eq. (3.35), we have,

$$\begin{aligned}
\dot{V} &= [e'_y \quad e''_y] \begin{bmatrix} K_{Py} + k_y & \beta m_y \dot{x}^2 \\ \beta m_y \dot{x}^2 & m_y \dot{x}^2 \end{bmatrix} \begin{bmatrix} e_y \\ e'_y \end{bmatrix} \\
&+ [e_y \quad \sigma_y] \begin{bmatrix} \alpha & K_{Ly} \\ K_{Ly} & \beta K_{Ly} \end{bmatrix} \begin{bmatrix} e_y \\ \sigma_y \end{bmatrix} \\
&= (K_{Py} + k_y) e_y e'_y + (e'_y + \beta e_y) m_y \dot{x}^2 e''_y \\
&+ \beta m_y \dot{x}^2 e'^2_y + \alpha e_y e'_y + K_{Ly} e'_y \sigma_y + \beta K_{Ly} e_y \sigma_y \\
&+ K_{Ly} e_y^2 \\
&= (K_{Py} + k_y + \alpha) e_y e'_y + \beta m_y \dot{x}^2 e'^2_y \\
&+ (e'_y + \beta e_y) K_{Ly} \sigma_y + K_{Ly} e_y^2 \\
&+ (e'_y + \beta e_y) (\rho - (m_y \ddot{x} + c_y \dot{x} + K_{Dy})) e'_y \\
&- (K_{Py} + k_y) e_y - K_{Ly} \sigma_y \\
&= -(m_y \ddot{x} + c_y \dot{x} + K_{Dy} - \beta m_y \dot{x}^2) e'^2_y \\
&- (\beta (K_{Py} + k_y) - K_{Ly}) e_y^2 \\
&- (\beta (m_y \ddot{x} + c_y \dot{x} + K_{Dy}) - \alpha) e_y e'_y \\
&+ (e'_y + \beta e_y) \rho
\end{aligned} \tag{3.36}$$

According to Eq. (3.27), if we choose  $K_{Dy} > \frac{\alpha}{\beta} + m_y \|\ddot{x}\| + c_y \|\dot{x}\|$ , we can prove that,

$$\begin{aligned}
&\beta (m_y \ddot{x} + c_y \dot{x} + K_{Dy}) - \alpha > \beta \{m_y (\ddot{x} + \|\ddot{x}\|) + c_y (\dot{x} + \|\dot{x}\|)\} \\
&\geq 0
\end{aligned} \tag{3.37}$$

Using the following inequality,  $-e_y e'_y \leq \frac{1}{2} (e_y^2 + e'^2_y)$ , Eq. (3.36) can be rewritten as,

$$\begin{aligned}
\dot{V} \leq & - \left( \left( 1 - \frac{\beta}{2} \right) (m_y \ddot{x} + c_y \dot{x} + K_{Dy}) - \beta m_y \dot{x}^2 \right) e_y'^2 \\
& - \beta \left( K_{Py} + k_y - \frac{K_{Ly}}{\beta} - \frac{m_y \ddot{x} + c_y \dot{x} + K_{Dy}}{2} \right. \\
& \left. + \frac{\alpha}{2\beta} \right) e_y^2 + (e_y' + \beta e_y) \rho
\end{aligned} \tag{3.38}$$

From Eq. (3.27), if we choose  $K_{Py} > \frac{K_{Ly}}{\beta} + \frac{1}{2}(m_y \|\ddot{x}\| + c_y \|\dot{x}\| + K_{Dy})$ , one can prove that,

$$\begin{aligned}
\rho_e &= K_{Py} + k_y + \frac{\alpha}{2\beta} - \frac{m_y \ddot{x} + c_y \dot{x} + K_{Dy}}{2} - \frac{K_{Ly}}{\beta} \\
&\geq k_y + \frac{\alpha}{2\beta} + \frac{1}{2} \{ m_y (\|\ddot{x}\| - \ddot{x}) + c_y (\|\dot{x}\| - \dot{x}) \} > 0
\end{aligned} \tag{3.39}$$

Similarly, from Eq. (3.27), if we choose  $K_{Dy} > \frac{\alpha}{\beta} + m_y \|\ddot{x}\| + c_y \|\dot{x}\|$  and  $\frac{\alpha}{\beta} > \frac{2\beta}{1-\beta} m_y \|\ddot{x}\|^2$ , then we have,

$$\rho_{e'} = \left( 1 - \frac{\beta}{2} \right) (m_y \ddot{x} + c_y \dot{x} + K_{Dy}) - \beta m_y \dot{x}^2 + \frac{\alpha}{2} > 0 \tag{3.40}$$

Applying Eq. (3.39) and Eq. (3.40) to Eq. (3.38), we obtain,

$$\begin{aligned}
\dot{V} &\leq -\beta \rho_e e_y^2 - \rho_{e'} e_y'^2 + (\beta e_y + e_y') \rho \\
&\leq -\beta \rho_e e_y^2 + \beta e_y |\rho|_{\max} - \rho_{e'} e_y'^2 + e_y' |\rho|_{\max}
\end{aligned} \tag{3.41}$$

Applying another inequality,

$$az - bz^2 \leq \frac{a^2}{b} - \frac{1}{4}bz^2 \tag{3.42}$$

For  $a > 0$  and  $b > 0$ , we have,

$$\begin{cases} \beta|\rho|_{\max}e_y - \beta\rho_e e_y^2 \leq -\frac{\beta\rho_e}{4}e_y^2 + \frac{\beta|\rho|_{\max}^2}{\rho_e} \\ |\rho|_{\max}e_y' - \rho_{e'} e_y'^2 \leq \frac{\rho_{e'}}{4}e_y'^2 + \frac{|\rho|_{\max}^2}{\rho_{e'}} \end{cases} \quad (3.43)$$

Applying Eq. (3.43) to Eq.(3.41), we get,

$$\dot{V} \leq -\frac{\beta\rho_e}{4}e_y^2 - \frac{\rho_{e'}}{4}e_y'^2 + \left(\frac{\beta}{\rho_e} + \frac{1}{\rho_{e'}}\right)|\rho|_{\max}^2 \quad (3.44)$$

Therefore, according to Lemma 1, we can demonstrate that both the contour tracking error and the derivative of the contour tracking error are bounded as follows,

$$\begin{cases} \|e_y\| \leq 2\sqrt{\frac{1}{\rho_e^2} + \frac{1}{\beta\rho_e\rho_{e'}}}|\rho|_{\max} \\ \|e_y'\| \leq 2\sqrt{\frac{\beta}{\rho_e\rho_{e'}} + \frac{1}{\rho_{e'}^2}}|\rho|_{\max} \end{cases} \quad (3.45)$$

According to Eq. (3.45), one can see that the contour error and its derivative are bounded. From Eq. (3.45), it is also shown that the maximum errors can be reduced to very small values by increasing control gain  $K_{Py}$  (related to  $\rho_e$ ) and  $K_{Dy}$  (related to  $\rho_{e'}$ ). From Eq. (3.39) and Eq. (3.40), one can see that a large constant  $\alpha$  will also increase parameters  $\rho_e$  and  $\rho_{e'}$ . Therefore, the tracking errors will be reduced according to Eq. (3.45). Control gain  $K_{Iy}$  has an indirect contribution for the control of final tracking error by increasing the constant parameter  $\alpha$  from Eq. (3.27).

### 3.6 POSITION DOMAIN CONTROL WITH CROSS-COUPLED CONTROL

The CCC control laws use in this research is derived from (Koren & Lo, Variable-Gain Cross-Coupling Controller for Contouring, 1991). For linear contour tracking, the contour error and its derivative are derived from the tracking error of the  $j$  and  $k$  axes as

$$\begin{aligned} e_c &= -C_j e_j + C_k e_k \\ e_c' &= -C_j e_j' + C_k e_k' \end{aligned} \quad (3.46)$$

Where  $C_j = \sin \theta$  and  $C_k = \cos \theta$  refers to the variable-gains in the cross-coupled control. For a linear motion, both  $C_j$  and  $C_k$  are constant since  $\theta$  is constant. The corresponding circular motion contour errors are defined as:

$$\begin{aligned} e_c &= -C_j e_j + C_k e_k \\ e_c' &= -C_j e_j' + C_k e_k' - C_j' e_j + C_k' e_k \end{aligned} \quad (3.47)$$

Where  $C_j = \sin \theta - \frac{e_j}{2R}$ ,  $C_k = \cos \theta + \frac{e_k}{2R}$ ,  $C_j' = \dot{\theta} \cos \theta - \frac{e_j'}{2R}$ , and  $C_k' = -\dot{\theta} \sin \theta + \frac{e_k'}{2R}$ .

Here,  $R$  is the radius of the contour. From these relationships, it is possible to develop the control law in position domain to include cross-coupling control. The PDC controller law with CCC is given as

$$T_j(x) = K_P^j e_i(x) + K_I^j \int_0^x e_j(s) ds + K_D^j e_j'(x) - C_j (K_P^j e_c + K_D^j e_c') \quad (3.48)$$

And

$$T_k(x) = K_P^k e_k(x) + K_I^k \int_0^x e_k(s) ds + K_D^k e_k'(x) + C_k (K_P^k e_c + K_D^k e_c') \quad (3.49)$$

Where axes  $j$  and  $k$  are the slaves axes in the position domain with the  $x$ -axis as the reference axis. Both Eq. (3.48) and Eq. (3.49) are equivalent to the PDC control law of Eq. (3.19) except with the addition of the cross-couple elements.

## Chapter 4 : SIMULATION RESULTS

### 4.1 SIMULATION SETUP

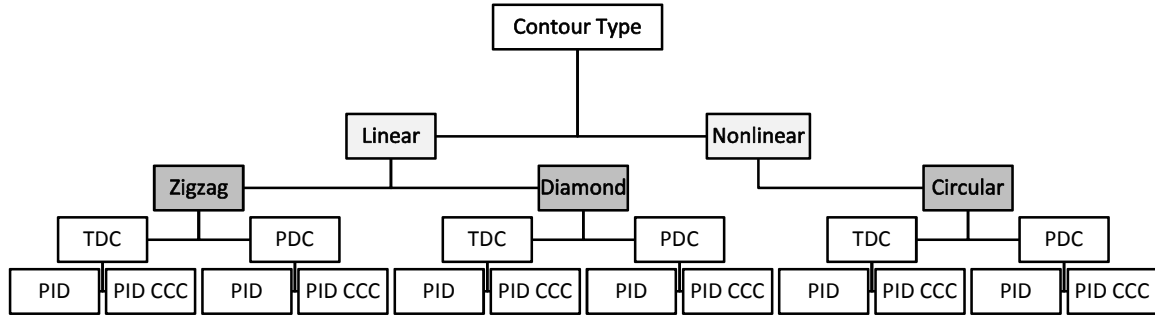
A virtual CNC system that will be utilized for the simulation is a 3 DOF system as shown in Figure 3-1. The dynamic parameters for each axis of the system are listed in Table 1. These parameters are non-mismatching to simplify the initial comparative analysis between each control laws. The sampling rate of the TDC and the equidistant sampling of the reference axis for the PDC are listed in Table 2. Note that the PDC sampling for the linear and circular contour are different to account for the effects of the nonlinear contour. The total number of simulations is outlined in Figure 4-1.

**Table 1: Non-mismatching System dynamic parameters**

Axis	X	Y	Z
Mass ( $m_i$ [ $kg$ ])	1	1	1
Damping ( $c_i$ [ $Ns/m$ ])	7	7	7
Spring constant ( $k_i$ [ $N/m$ ])	50	50	50

**Table 2: TDC sampling rate and PDC equidistant sampling**

TDC sampling rate	1000 Hz
PDC sampling (linear contour)	0.005 m
PDC sampling (circular contour)	0.002 m



**Figure 4-1: Simulation outline**

As indicated in Figure 4-1, both linear and nonlinear contour types will be simulated. For the linear contour type, the simulation will consist of a zigzag and a diamond contour. For the nonlinear contour type, the simulation will consist of a circular contour. For all types of contours, a position domain PID controller (referred to as a PDC) as well as a time domain PID (referred to as a TDC) will be used to control the motion. Furthermore, a CCC component will also be added to both TDC and PDC controllers to compare their performance without CCC.

## 4.2 TRAJECTORY PLANNING

To ensure a smooth trajectory in all axes, a high-order polynomial is used to define their position, velocity, and acceleration with respect to time (Craig & J, 1989). The high-ordered polynomial is given as,

$$r(t) = 10 \left( \frac{t}{T} \right)^3 - 15 \left( \frac{t}{T} \right)^4 + 6 \left( \frac{t}{T} \right)^5 \quad (4.1)$$

With the first and second derivative being,

$$\begin{cases} \dot{r}(t) = \frac{1}{T} \left\{ 30 \left( \frac{t}{T} \right)^2 - 60 \left( \frac{t}{T} \right)^3 + 30 \left( \frac{t}{T} \right)^4 \right\} \\ \ddot{r}(t) = \frac{1}{T^2} \left\{ 60 \left( \frac{t}{T} \right) - 180 \left( \frac{t}{T} \right)^2 + 120 \left( \frac{t}{T} \right)^3 \right\} \end{cases} \quad (4.2)$$

where  $t$  represents the time while  $T$  is the total time of the motion from an initial to a final position. Therefore, the axial position of the  $i^{th}$  axis is defined as,

$$q_i(t) = (p_f - p_i)r(t) \quad (4.3)$$

where  $p_f$  and  $p_i$  are the final and initial position respectively.

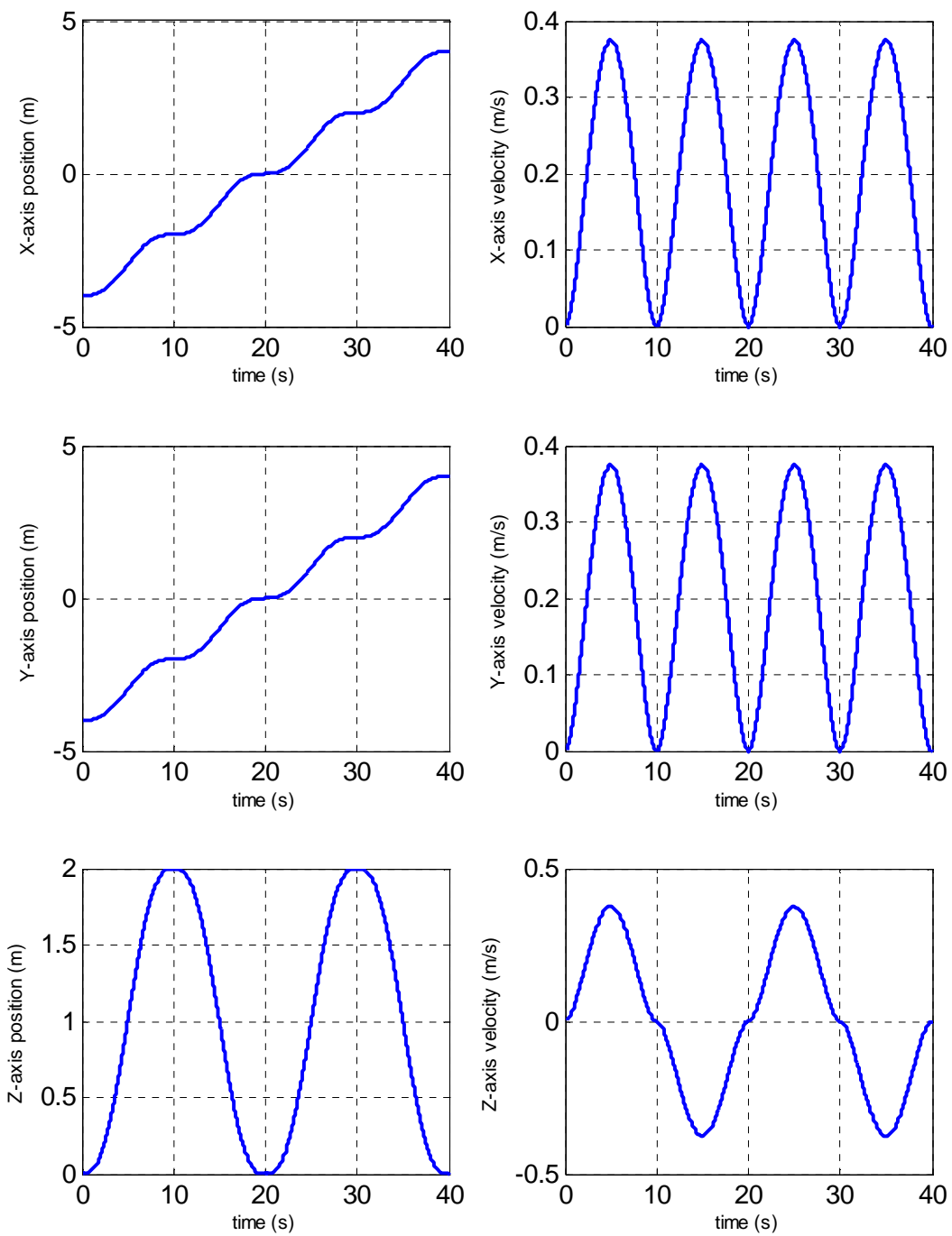
The initial and final axial positions for each type of motion in the simulation are listed in Table 3. Here, the linear contours (zigzag and diamond) are divided into four linear segments, while the circular contour has two semi-circle segments. The total motion time for each segment are displayed in Table 4. The transition between each segment is a stop-and-go motion rather than continuous. Although both types of transitions are applicable in modern CNC machines, the latter is preferred as it reduces the overall processing time. This thesis will utilize the stop-and-go transition for simplicity. The plots for the desired axial trajectories for position and velocity for zigzag, diamond, and circular motions in time domain are shown in Figure 4-2, Figure 4-3, and Figure 4-4 respectively.

**Table 3: Initial and final axial positions for linear and circular contour simulations ( $p_i$  and  $p_f$  denote initial and final position respectively.)**

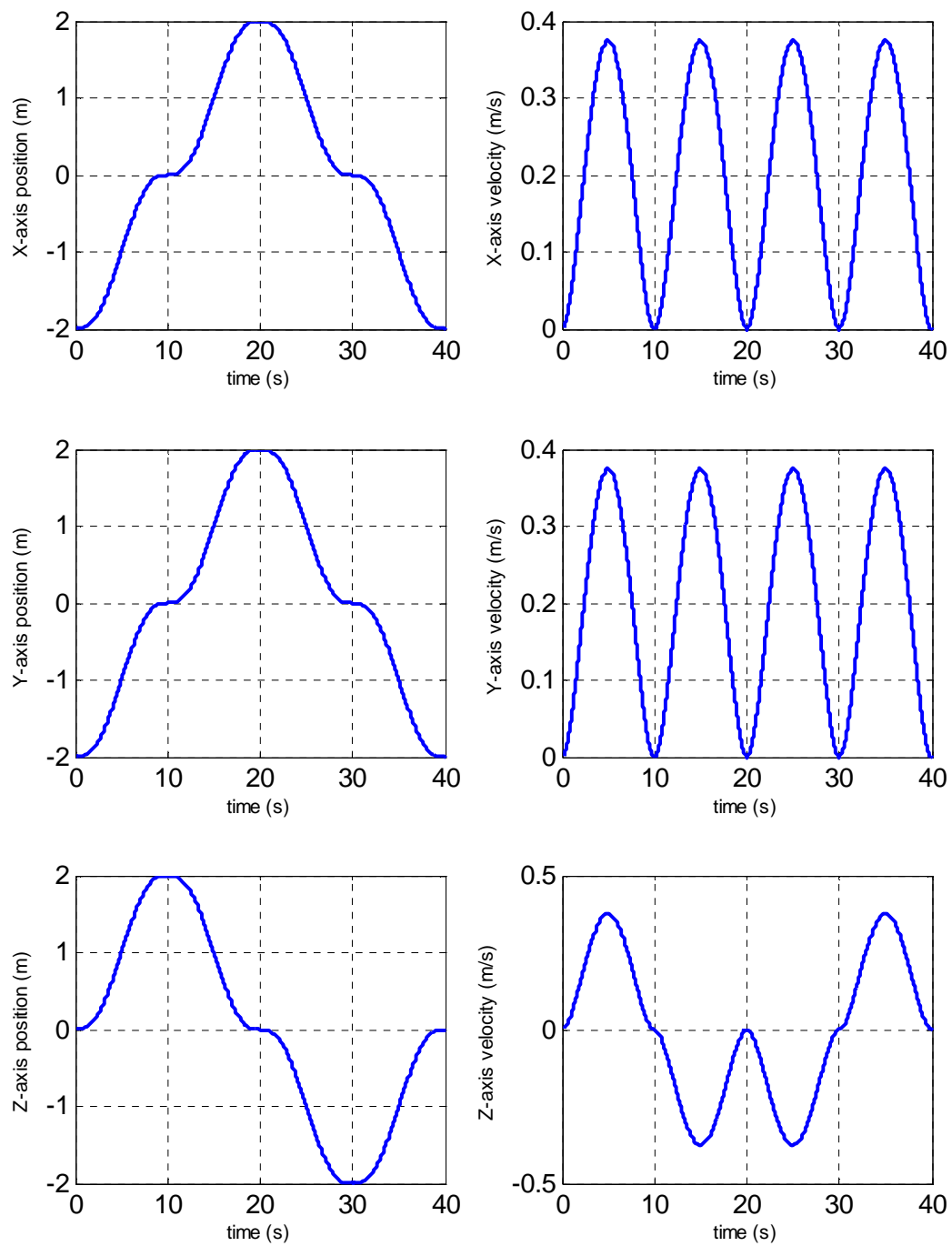
	Segment 1		Segment 2		Segment 3		Segment 4	
	$p_i(\text{m})$	$p_f(\text{m})$	$p_i(\text{m})$	$p_f(\text{m})$	$p_i(\text{m})$	$p_f(\text{m})$	$p_i(\text{m})$	$p_f(\text{m})$
<i>Zigzag contour</i>								
<i>x-axis</i>	-4	-2	-2	0	0	2	2	4
<i>y-axis</i>	-4	-2	-2	0	0	2	2	4
<i>z-axis</i>	0	2	2	0	0	2	2	0
<i>Diamond contour</i>								
<i>x-axis</i>	-2	0	0	2	2	0	0	-2
<i>y-axis</i>	-2	0	0	2	2	0	0	-2
<i>z-axis</i>	0	2	2	0	0	-2	-2	0
<i>Circular contour</i>								
<i>x-axis</i>	-0.5	0.5	0.5	-0.5				
<i>y-axis</i>	0	0	0	0				
<i>z-axis</i>	0	0	0	0				

**Table 4: Segment and total motion time**

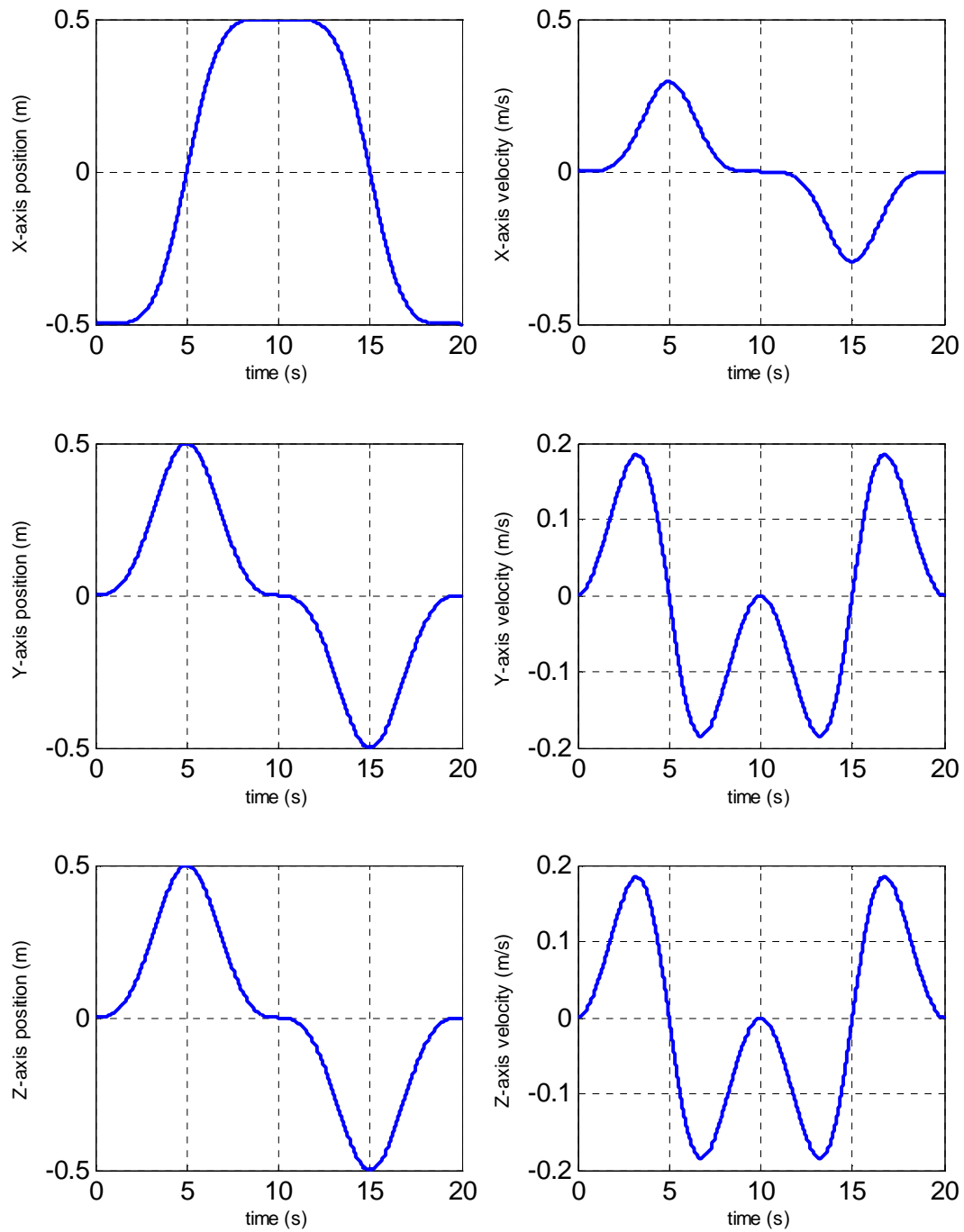
	Zigzag	Diamond	Circular
Segment motion time (s)	10	10	10
Total motion time (s)	40	40	20



**Figure 4-2: Desired trajectories for zigzag contour axial positions and velocities**



**Figure 4-3: Desired trajectories for diamond contour axial positions and velocities**



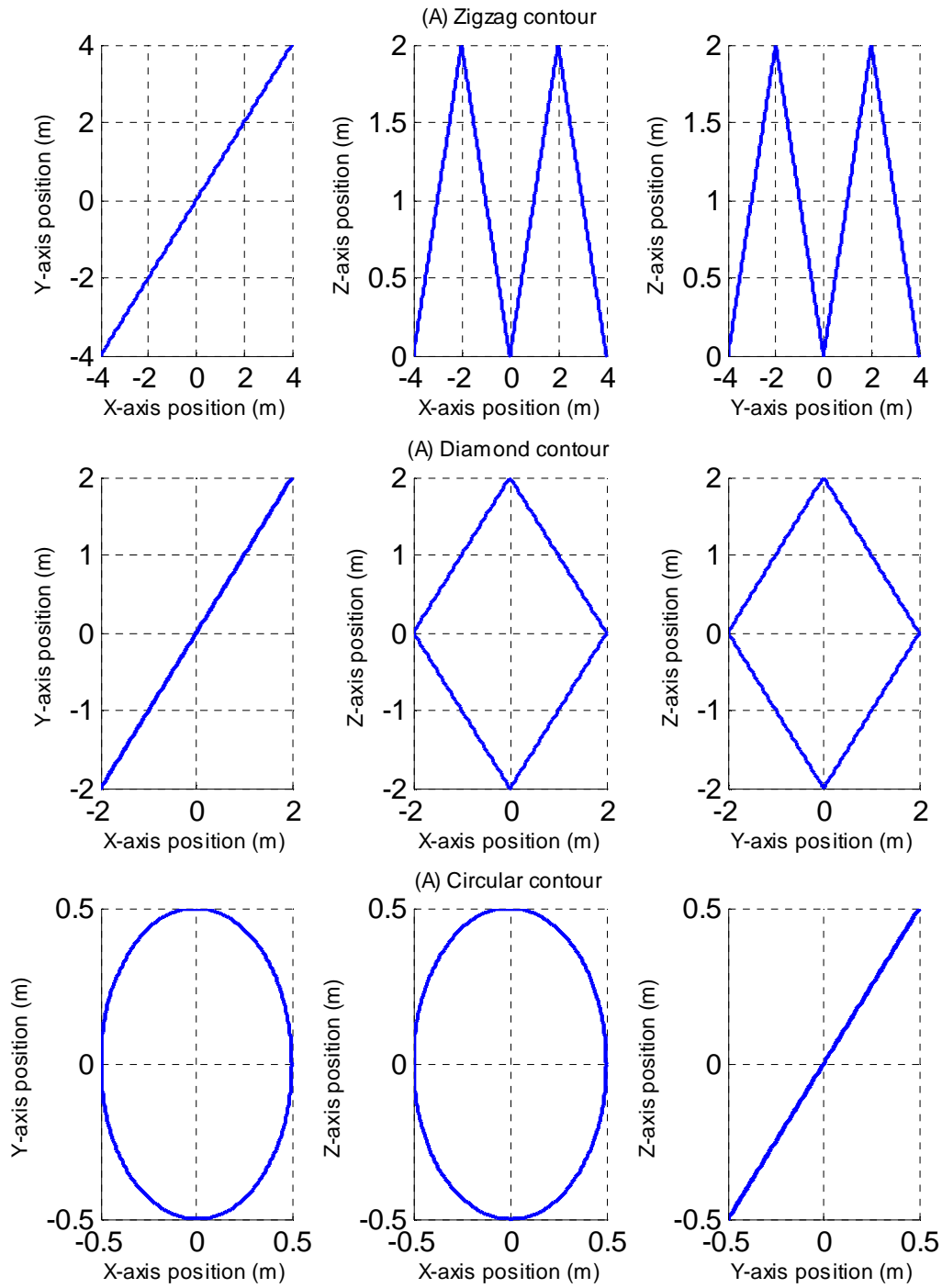
**Figure 4-4: Desired trajectories for circular contour axial positions and velocities**

From the above figures, the maximum axial velocity for each contour is listed in Table 5. Since all three axes in the linear motions travel the same distance throughout the entire motion, the absolute maximum axial velocity in all axes are the same. In circular motion, the  $x$ -axis travels the greatest distance within the same time period relative to the other axes, therefore its performance is the limiting factor. For linear motion, the maximum feed rate is  $0.375\text{ m/s}$ , while for circular motion, the maximum feed rate is  $0.295\text{ m/s}$ . These values are within modern CNC machine performance specifications.

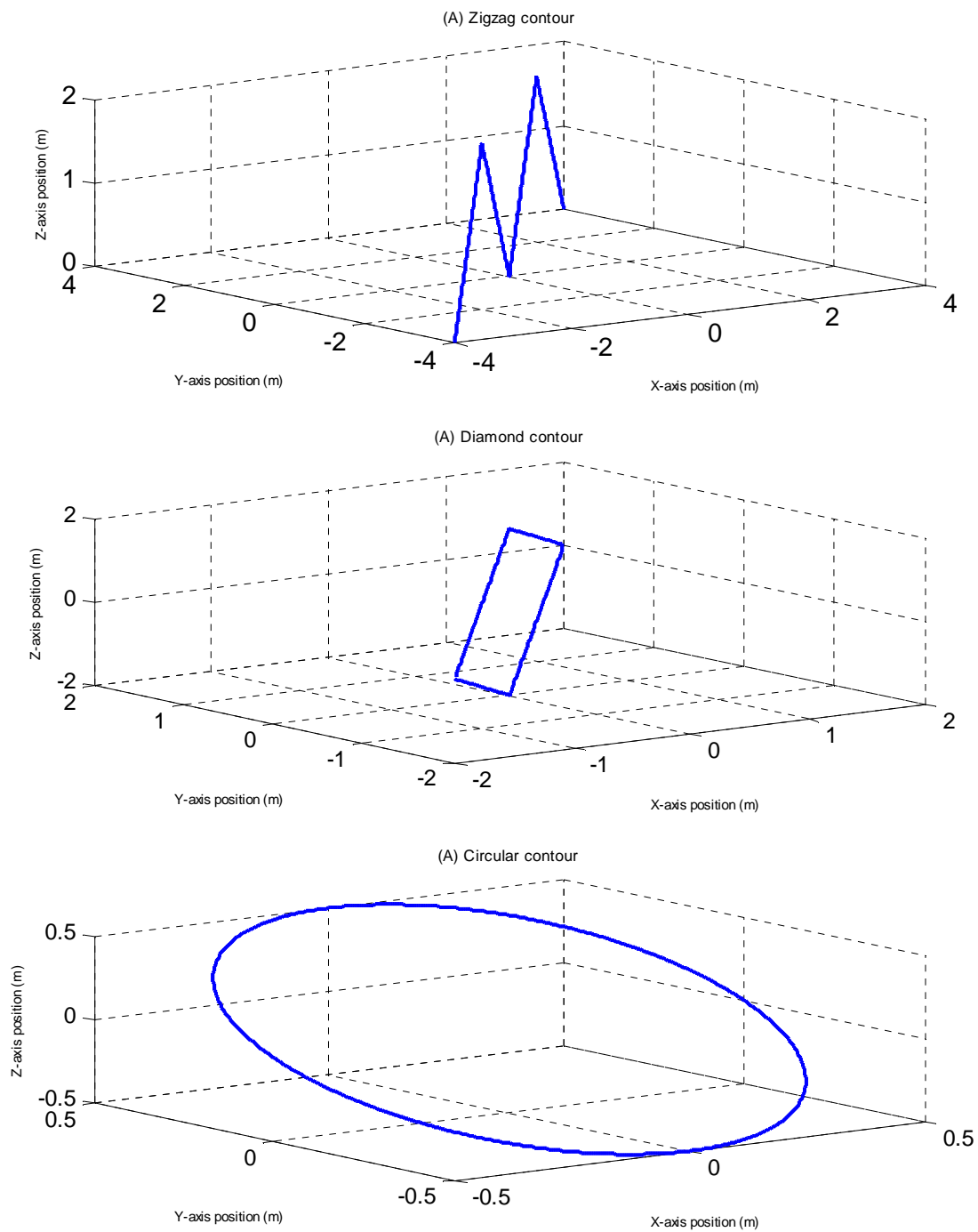
**Table 5: Maximum absolute axial velocities**

	<b>x-axis (<math>m/s</math>)</b>	<b>y-axis (<math>m/s</math>)</b>	<b>z-axis (<math>m/s</math>)</b>
<b>Zigzag</b>	0.375	0.375	0.375
<b>Diamond</b>	0.375	0.375	0.375
<b>Circular</b>	0.295	0.184	0.184

Figure 4-5 and Figure 4-6 below are the plots of the desired linear and circular contours in 2D and 3D respectively.



**Figure 4-5: Desired (A) Zigzag, (B) Diamond and (C) Circular contours in planar views**



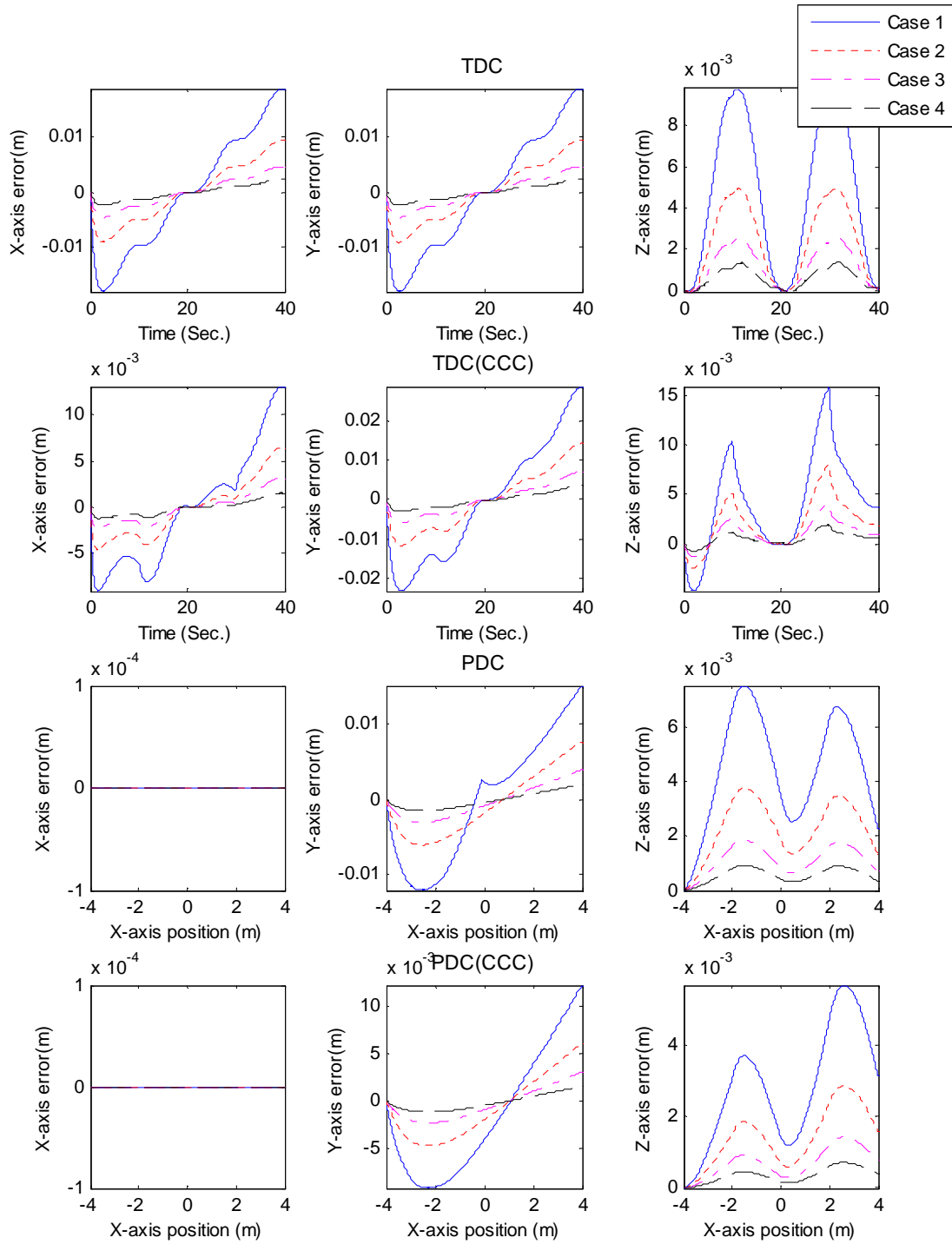
**Figure 4-6: Desired (A) Zigzag, (B) Diamond and (C) Circular contours in 3D views**

### 4.3 GAIN SELECTION

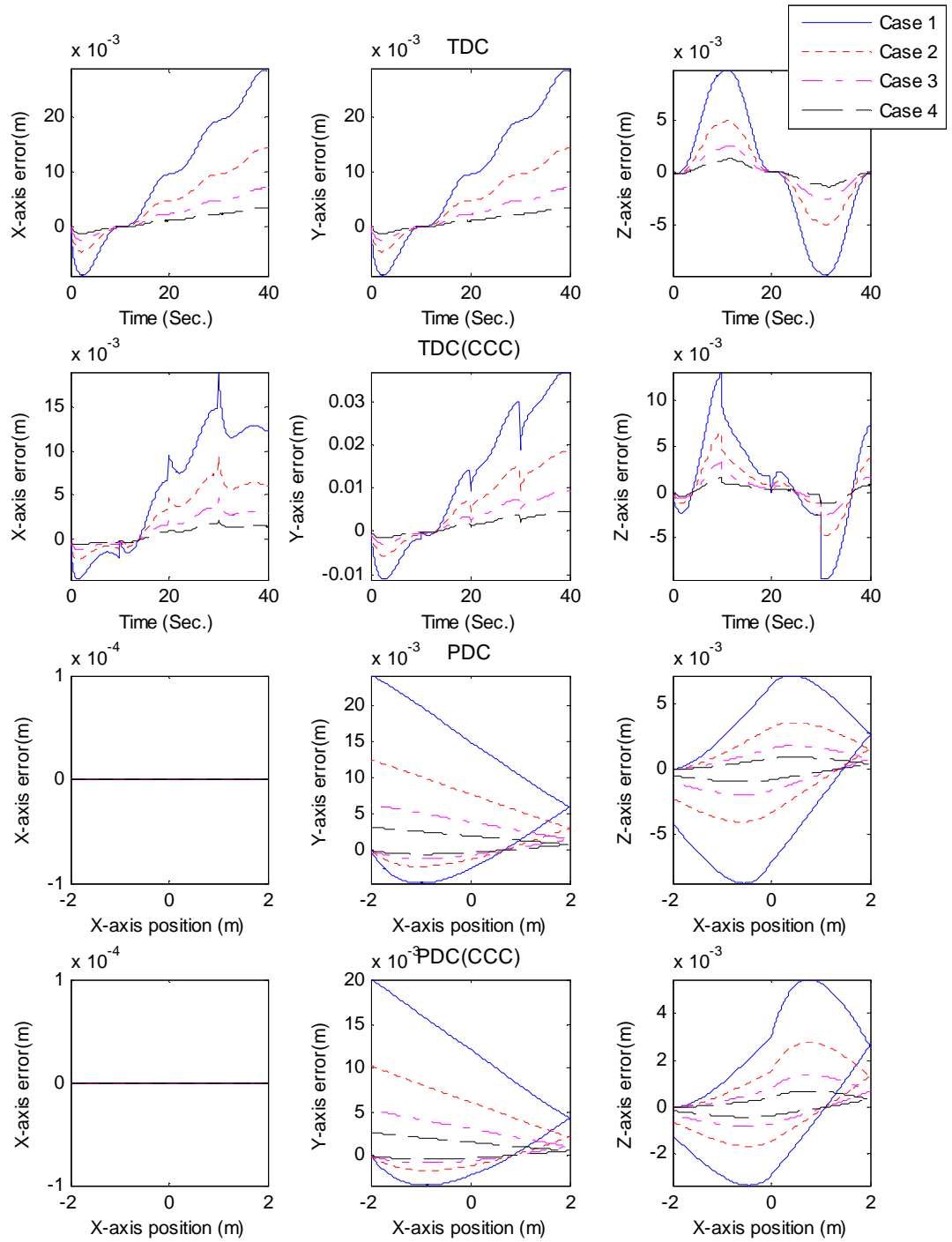
The selection of control gains for any system is highly important to its overall performance. For this simulation, the gains were selected based on a trial and error method with the main objective being selecting the combination of control gains that resulted in the lowest axial tracking error without overshoot. The controllers gains used for the simulation are listed Table 6. Results for **Case 4** will be presented in this section as it provided the sufficient performance in terms of relatively low tracking errors and contouring errors (on the order of  $10^{-3}m$  or lower) for all three simulated contours. The same control gains will be utilized in both time and position domain controllers for all contours. From Figure 4-7 to Figure 4-9, it is evident that as the controller gains increases (from **Case 1** to **Case 4**), improvement in the axial tracking is observed for both the linear and circular contours with the TDC and PDC. This means that both controllers are stable within the specified range of the controller gains.

**Table 6: Linear contour controller gains for TDC, PDC, and CCC**

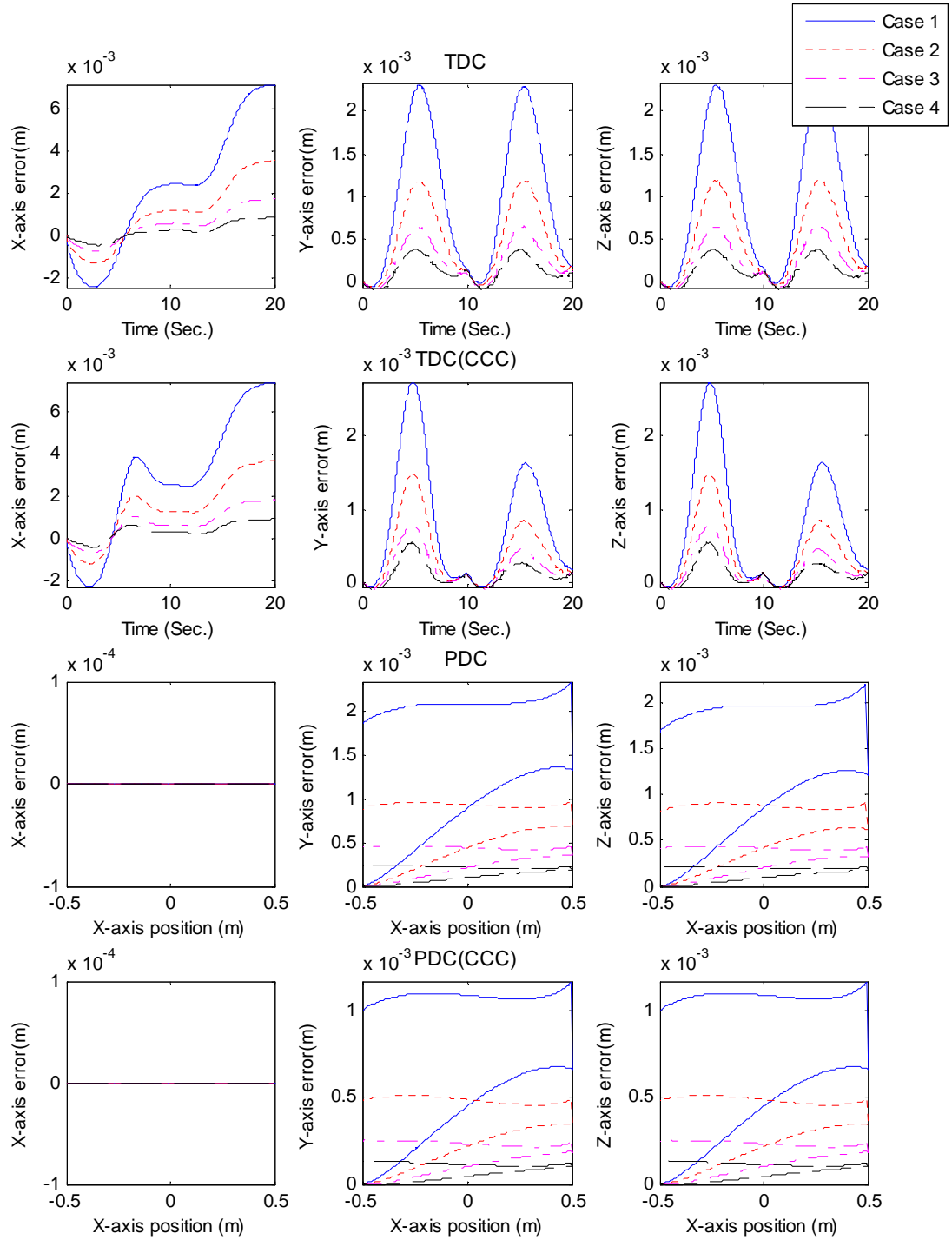
	<b>P</b>	<b>I</b>	<b>D</b>	<b>P<sub>ccc</sub></b>	<b>D<sub>ccc</sub></b>
<b>Case 1</b>	10000	6225	8750	10000	8750
<b>Case 2</b>	20000	12500	17500	20000	17500
<b>Case 3</b>	40000	25000	35000	40000	35000
<b>Case 4 (baseline)</b>	80000	50000	70000	80000	70000



**Figure 4-7: Zigzag contour tracking error variations with increasing controller gains**



**Figure 4-8: Diamond contour tracking error variations with increasing controller gains**



**Figure 4-9: Circular contour tracking error variations with increasing controller gains**

## 4.4 LINEAR CONTOUR TRACKING RESULTS

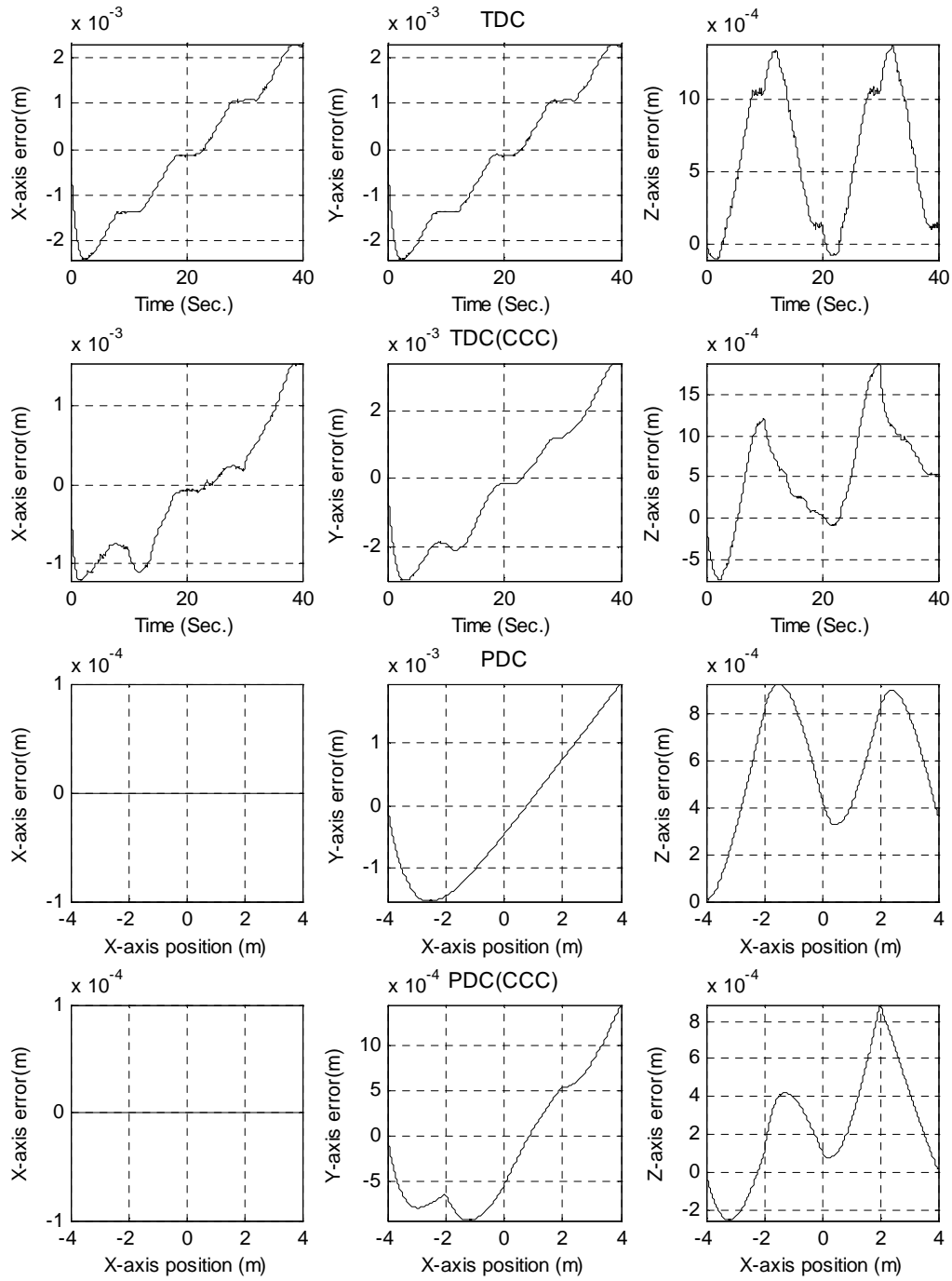
### 4.4.1 ZIGZAG CONTOUR

Using the baseline (**Case 4**) PID controller gains listed in Table 6, good axial tracking performance was achieved for all four controllers; TDC, TDC with CCC, PDC, and PDC with CCC in the simulation of the zigzag contour. The plots of the axial tracking errors for all controllers are displayed in Figure 4-10. Note that in the TDC plots in Figure 4-10, the  $x$ -axis represents time while in the PDC plots, the  $x$ -axis represents the  $x$ -axis position. Comparison between the performance of the TDC and the PDC indicate that the PDC provided better axial tracking in terms of lower or comparable values of mean tracking error and lower standard of deviation in the  $y$  and  $z$  slave axes. The  $x$ -axis in the PDC controller is the reference axis, thus it yields zero tracking error as indicated in Figure 4-10. The best overall performance was observed with the PDC controller with CCC as the standard of deviation for the tracking error was lower (ranging from 0.0003m to 0.0006m) in comparison to the TDC, TDC with CCC, as well as the PDC controller without CCC. The results for all axial mean tracking errors and standard of deviations for the zigzag contour are listed in Table 7.

**Table 7: Mean and standard deviation (S.D) for zigzag motion axial tracking error<sup>1</sup>**

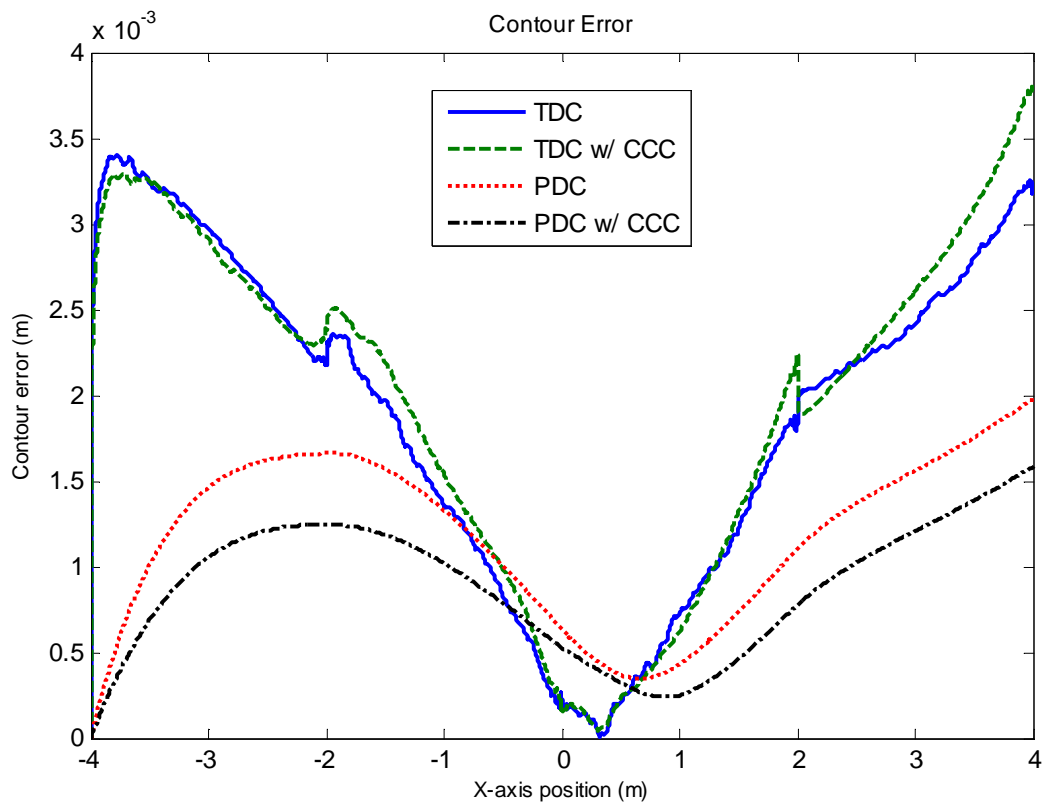
	X-axis		Y-axis		Z-axis	
	Mean(m)	S.D(m)	Mean(m)	S.D(m)	Mean(m)	S.D(m)
<b>TDC</b>	-0.000121	0.001376	-0.000121	0.001376	0.000604	0.000474
<b>TDC with CCC</b>	-0.000128	0.000786	-0.000190	0.001883	0.000527	0.000626
<b>PDC</b>	N/A	N/A	-0.000184	0.001077	0.000578	0.000258
<b>PDC with CCC</b>	N/A	N/A	-0.000140	0.000707	0.000238	0.000297

<sup>1</sup> X-axis tracking for both PDC controllers is not applicable since the  $x$ -axis is the reference axis therefore it yields zero tracking error.

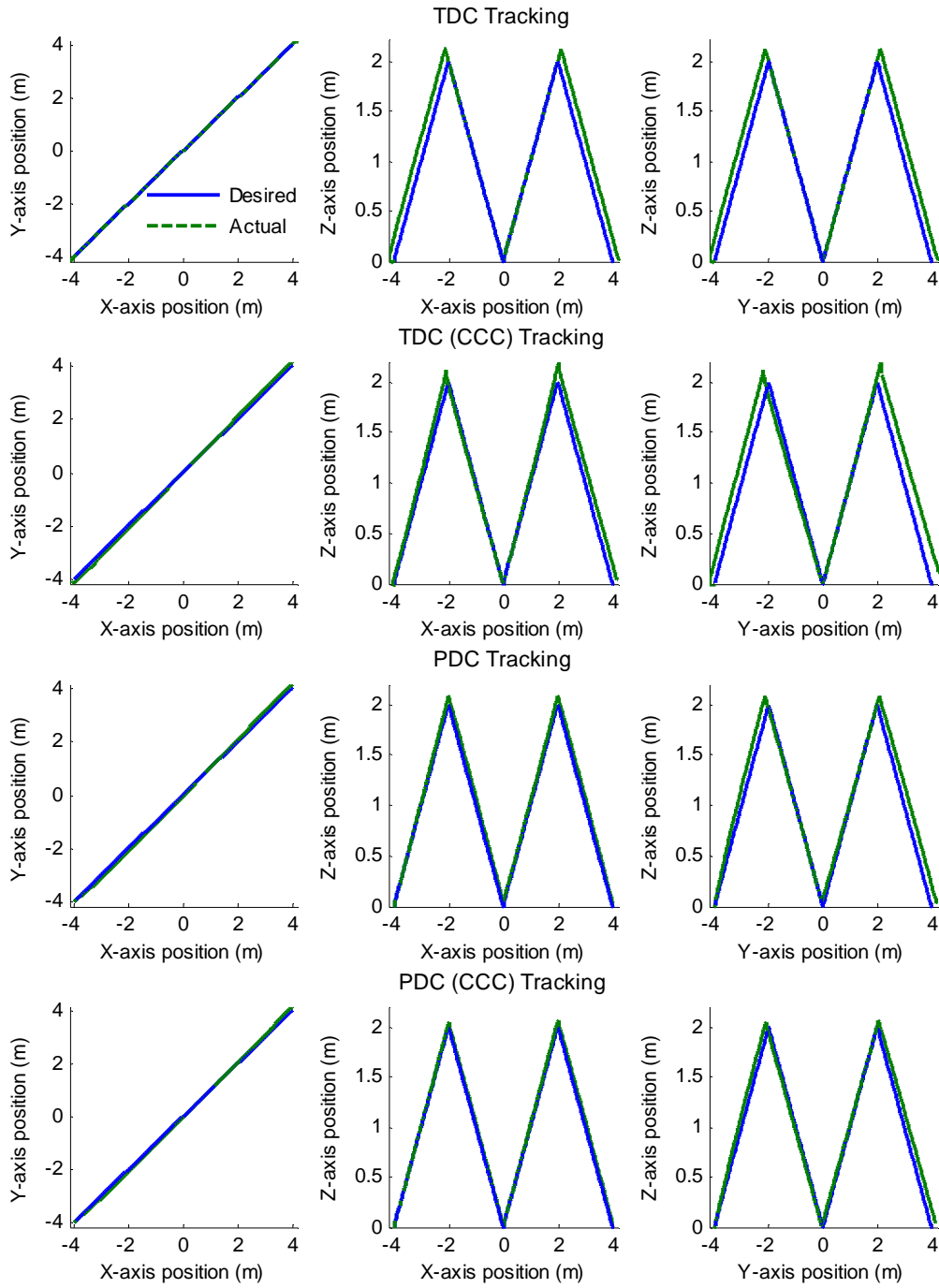


**Figure 4-10: Zigzag contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC**

In terms of the contouring accuracy performance, the PDC controller provided the highest contouring accuracy with lower values of mean contour error and standard of deviation over the equivalent TDC controller. The final value of the contour error for the PDC is approximately 50% lower in comparison to both TDC controllers. Furthermore, the addition of CCC to the PDC controller resulted in further improvement in the contouring performance over the other controllers. The plots of the contouring errors for all controllers are in Figure 4-11 while plot of the desired and actual contours are in Figure 4-12. Note that all errors in Figure 4-12 have been magnified by a factor of 100. From these figures, it is evident that the PDC with CCC provided the best contour performance with the least deviation from the desired contour.



**Figure 4-11: Zigzag motion contour error magnitude for TDC, TDC with CCC, PDC, and PDC with CCC**



**Figure 4-12: Zigzag contour tracking of TDC, TDC with CCC, PDC, and PDC with CCC<sup>2</sup>**

<sup>2</sup> The plots of all actual contours include an error magnification by a factor of 100

Table 8 lists the result for the contour performance of all controllers. Here, the mean contour error of the PDC with CCC is the lowest in comparison to all the other controllers. The improvement in the mean contour error ranged from 0.0004m to 0.0011m, with the highest improvement observed against both TDC controllers. A similar conclusion can be said for the standard of deviation of the contour error. Therefore, it can be concluded that the PDC with CCC provided the best tracking throughout the entire contour.

**Table 8: Magnitude of contour error mean, standard of deviation, and maximum**

	Mean(m)	S.D(m)	Max(m)
<b>TDC</b>	0.001840	0.001009	0.003401
<b>TDC with CCC</b>	0.001935	0.001068	0.003800
<b>PDC</b>	0.001168	0.000480	0.001976
<b>PDC with CCC</b>	0.000765	0.000283	0.001447

#### 4.4.2 DIAMOND CONTOUR

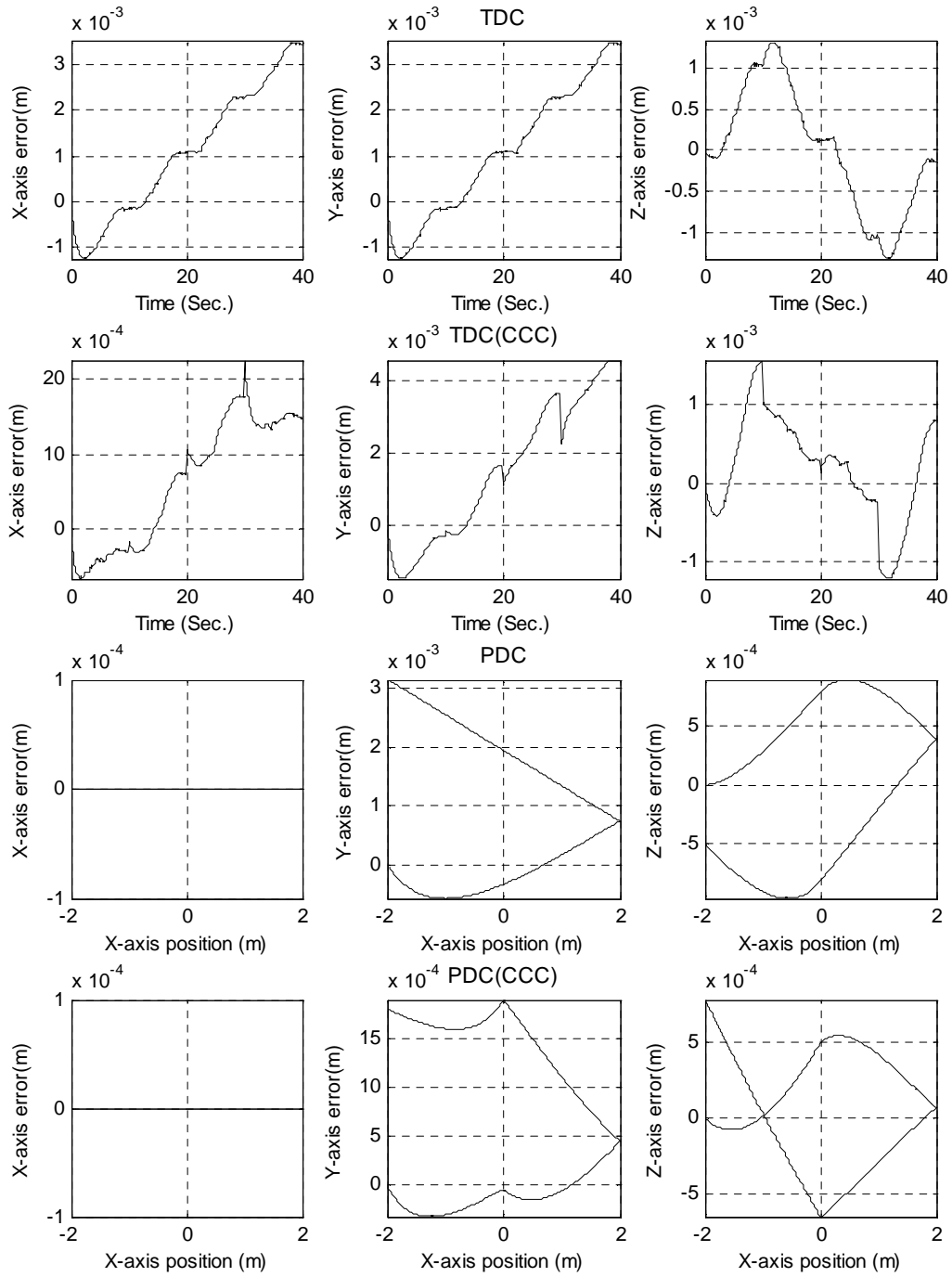
As with the zigzag motion, good tracking performance was achieved by all four controllers for the diamond contour type using **Case 4** controller gains with all axial tracking errors on the order of  $10^{-4}m$ . The plots of the axial tracking errors for each controller are shown in Figure 4-13. In the diamond motion simulation, both PDC controllers (with and without CCC) provided improved axial tracking performance for both the y and z slave axes with lower values of mean and standard of deviation of the tracking error over both TDC controllers. With the addition of CCC to the PDC controller, the performance of the original PDC controller is further improved with approximately a 30% reduction in the tracking error mean and standard of

deviation for the y-axis. However for the z-axis, a slight increased in the tracking error mean was observed with the addition of CCC. This does not necessarily mean that improvement in the z-axis tracking was not achieved as a 50% reduction in the standard of deviation of the tracking error was observed as indicated in Table 9.

**Table 9: Axial tracking error mean and standard deviation (S.D) for diamond motion<sup>3</sup>**

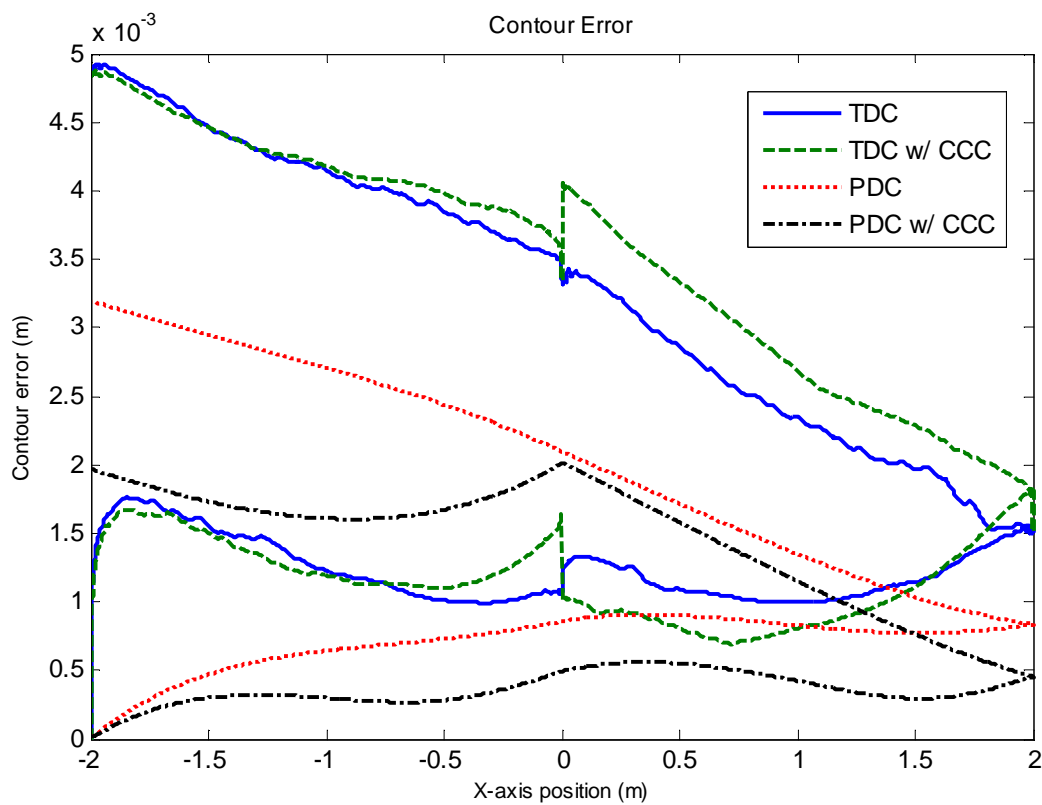
	<b>X-axis</b>		<b>Y-axis</b>		<b>Z-axis</b>	
	Mean(m)	S.D(m)	Mean(m)	S.D(m)	Mean(m)	S.D(m)
<b>TDC</b>	0.001080	0.001402	0.001080	0.001402	-0.000002	0.000759
<b>TDC with CCC</b>	0.000625	0.000830	0.001506	0.001928	0.000182	0.000658
<b>PDC</b>	N/A	N/A	0.000902	0.001185	0.000008	0.000631
<b>PDC with CCC</b>	N/A	N/A	0.000660	0.000816	0.000053	0.000352

<sup>3</sup> X-axis tracking for both PDC controllers is not applicable since the x-axis is the reference axis therefore it yields zero tracking error.

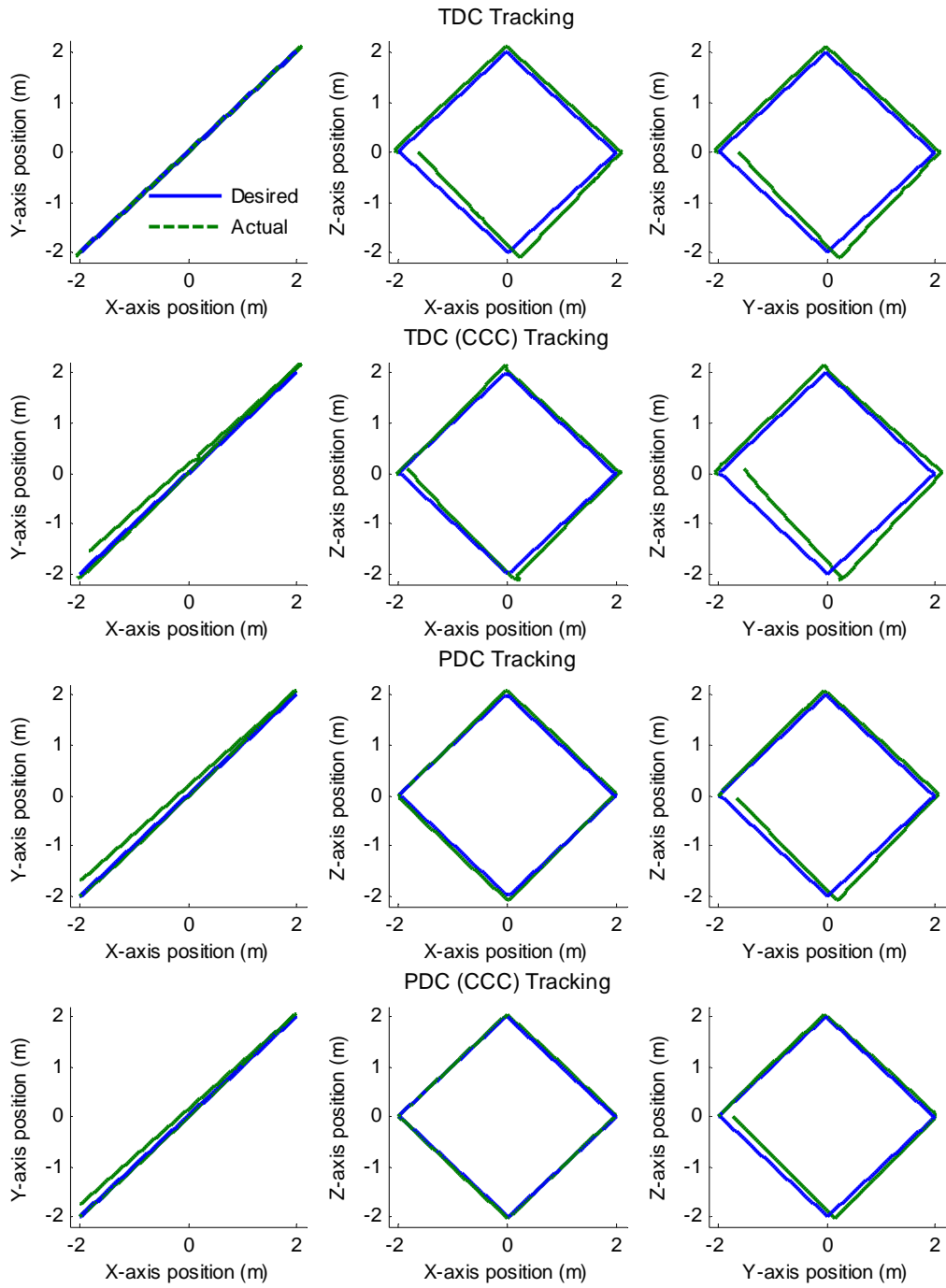


**Figure 4-13: Diamond contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC**

In terms of contouring accuracy, Figure 4-14 displays the contour errors for all four controllers. Here, the PDC with CCC resulted in a lowest contouring error throughout the entire contour. The final value of the contour error for the PDC was reduced by approximately 40% when compared to both TDC controllers. With the addition of the CCC, the error reduces further to 60%. Plots of the desired and actual contours in Figure 4-15 indicate that the PDC with CCC provided the best tracking over all the other controllers.



**Figure 4-14: Diamond motion contour error magnitude for TDC, TDC with CCC, PDC, and PDC with CCC**



**Figure 4-15: Diamond contour tracking of TDC, TDC with CCC, PDC, and PDC with CCC <sup>4</sup>**

<sup>4</sup> The plots of all actual contours include an error magnification by a factor of 100

Table 10 lists the results for the contouring error of all controllers. As with the zigzag contour, the PDC with CCC provided the lowest mean contouring error over all other controllers including the PDC without CCC in the diamond contour simulation. This represents approximately a 60% reduction in the mean contour error for the PDC with CCC controller over the both TDC controllers. A similar result can be concluded for the stand of deviation for contouring error.

**Table 10: Magnitude of contour error mean, standard of deviation, and maximum**

	Mean(m)	S.D(m)	Max(m)
<b>TDC</b>	0.002271	0.001296	0.004937
<b>TDC with CCC</b>	0.002385	0.001358	0.004874
<b>PDC</b>	0.001372	0.000856	0.003191
<b>PDC with CCC</b>	0.000908	0.000635	0.002016

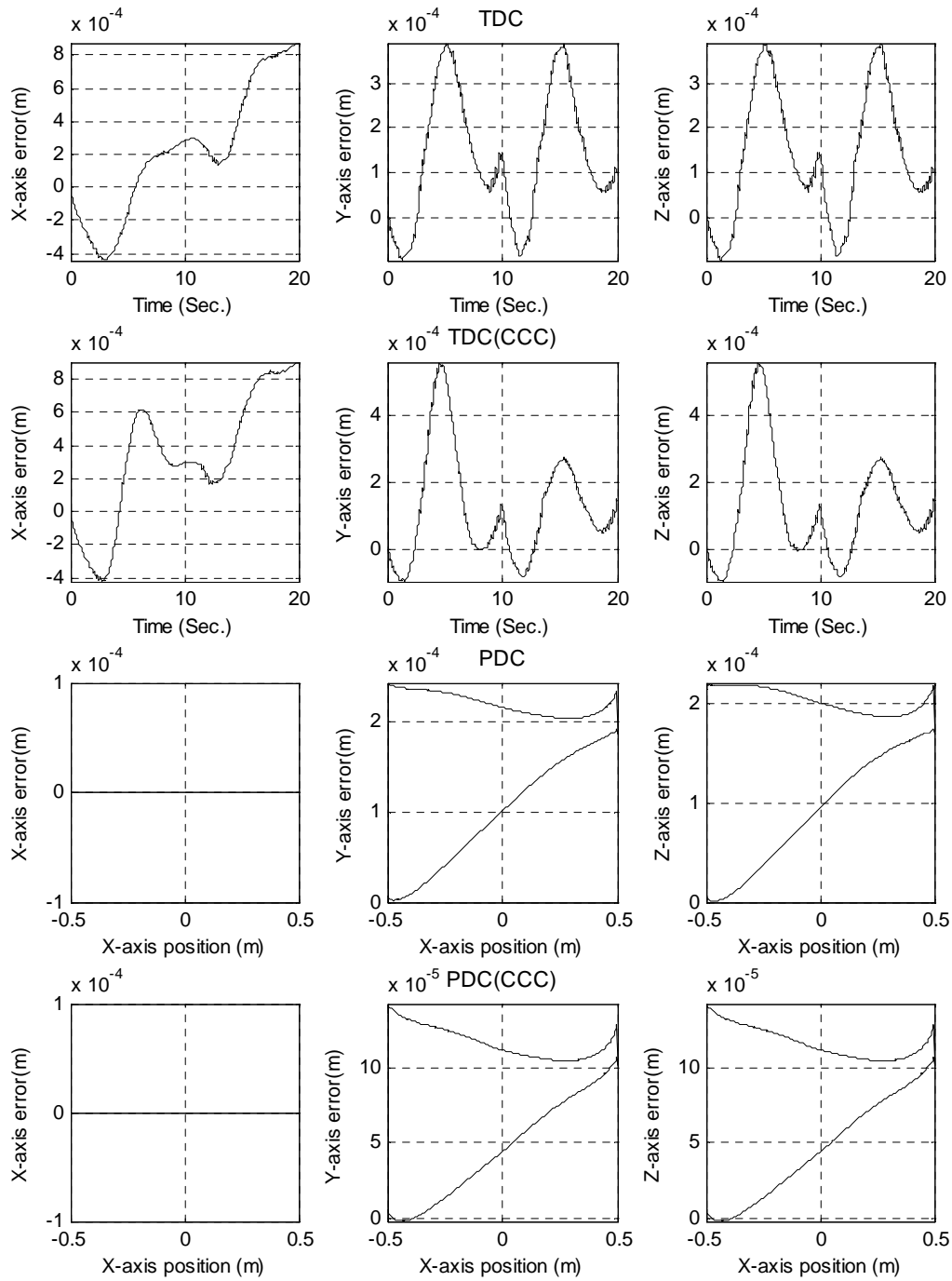
## 4.5 CIRCULAR CONTOUR RESULTS

For the circular contour simulation, good axial tracking performance was also achieved by all four controllers using the control gains specified in **Case 4**. The plot of the tracking errors for all controllers is shown in Figure 4-16. It can be seen that the final value of the tracking errors from all controllers are comparably low on the order of  $10^{-4}m$ . From Table 11, the standard of deviation in tracking error for the slave axes y and z are lower (by approximately 70%) when comparing the PDC with CCC over both TDC controllers while the mean of the tracking errors is approximately 30% lower. It can be concluded that the PDC with CCC provided the best tracking over the entire motion.

**Table 11: Axial tracking error mean and standard deviation (S.D) for circular motion<sup>5</sup>**

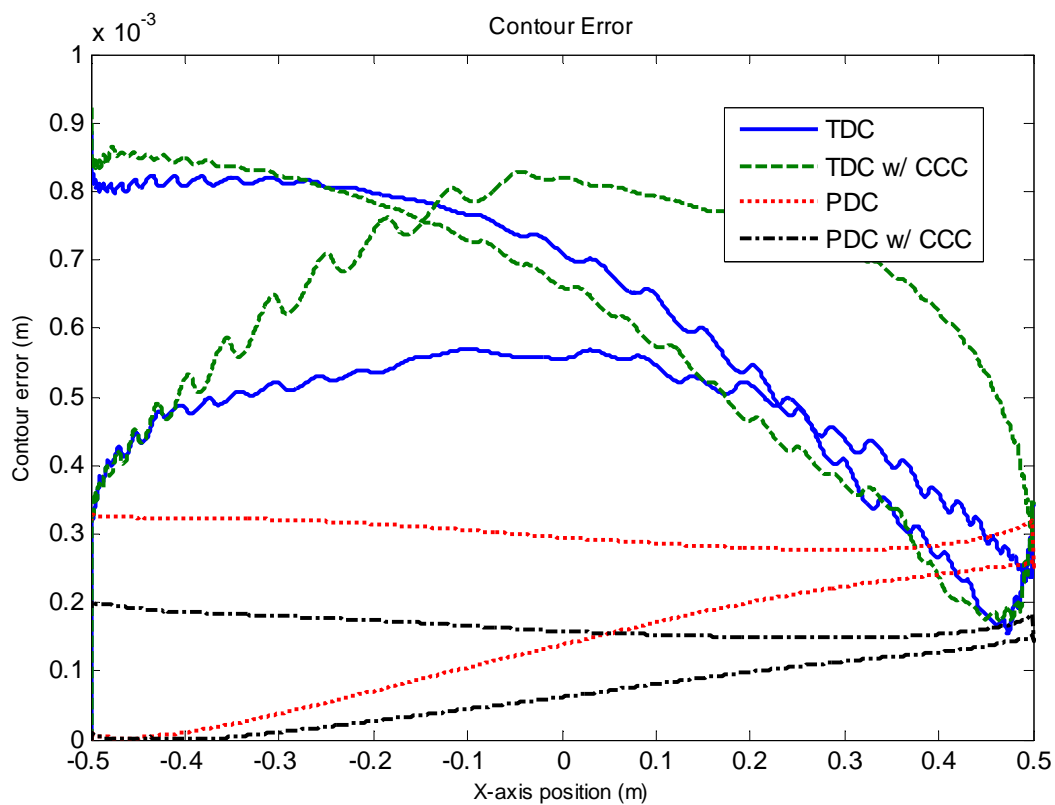
	X-axis		Y-axis		Z-axis	
	Mean(m)	S.D(m)	Mean(m)	S.D(m)	Mean(m)	S.D(m)
<b>TDC</b>	0.000212	0.000390	0.000135	0.000144	0.000135	0.000144
<b>TDC with CCC</b>	0.000326	0.000384	0.000123	0.000162	0.000123	0.000162
<b>PDC</b>	N/A	N/A	0.000158	0.000076	0.000147	0.000070
<b>PDC with CCC</b>	N/A	N/A	0.000081	0.000044	0.000081	0.000044

<sup>5</sup> X-axis tracking for both PDC controllers is not applicable since the x-axis is the reference axis therefore it yields zero tracking error.

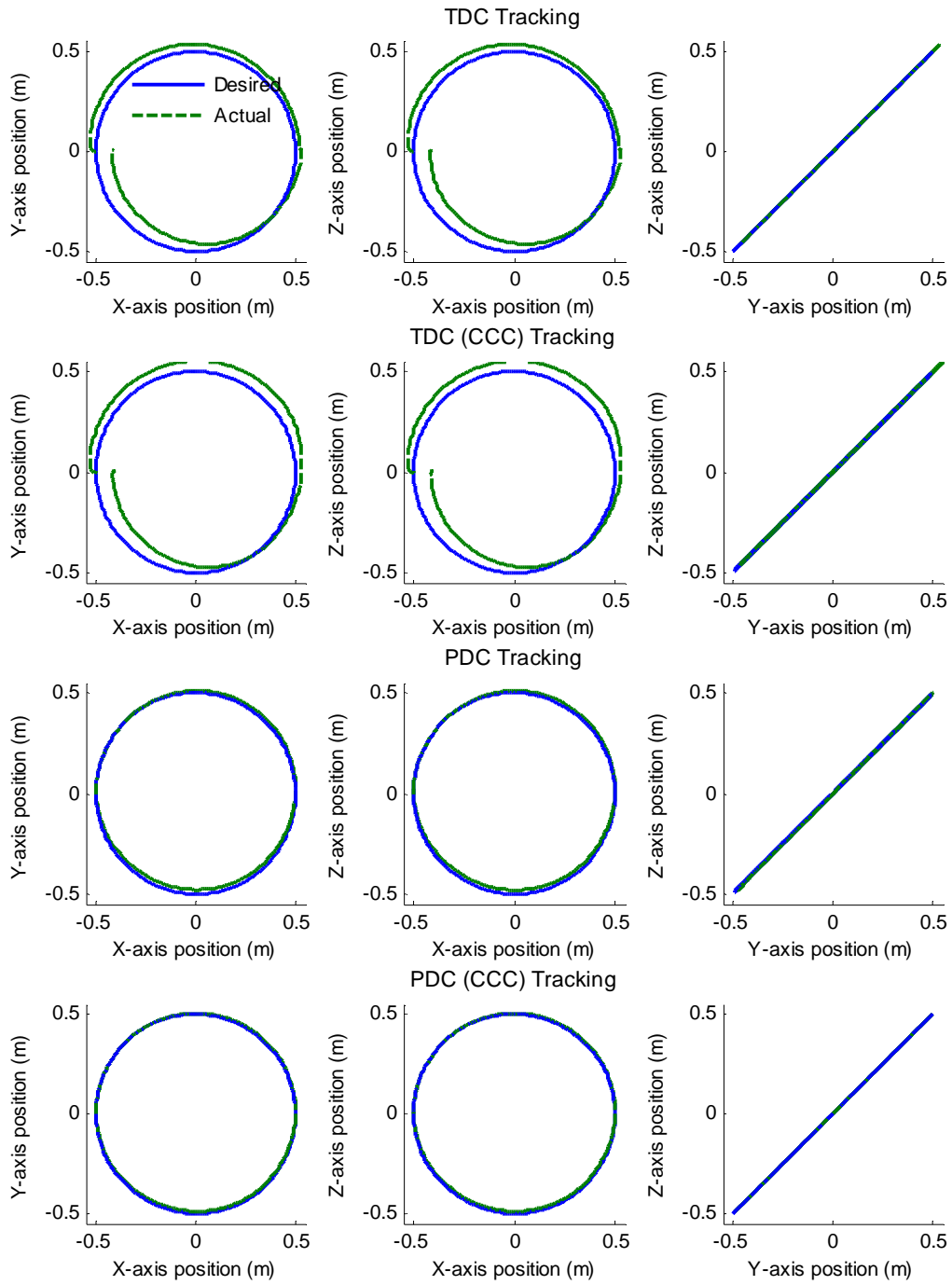


**Figure 4-16: Circular contour tracking error for TDC, PDC, and PDC with CCC**

Figure 4-17 displays the contouring error for all four controllers. The contouring error of both PDC controllers is lower than that of the TDC controllers. The final value of the contouring error is approximately 60% lower in the PDC controllers compared to the TDC controllers. The variation in the mean contour errors of both PDC controllers are also lower throughout the entire motion. Similarly to the linear motion simulation results, the PDC with CCC in circular motion simulation provided appreciable improvement in the contour tracking performance over the other control laws.



**Figure 4-17: Circular motion contour error for TDC, PDC, and PDC with CCC**



**Figure 4-18: Circular contour tracking of TDC, TDC with CCC, PDC, and PDC with CCC<sup>6</sup>**

<sup>6</sup> The plots of all actual contours include an error magnification by a factor of 100

From the desired contour and actual contour plot in Figure 4-18, it can be seen that both PDC controllers provided a smoother contour tracking over the entire motion, with the PDC with CCC performing better than the PDC without CCC in terms of smoothness in the motion tracking.

Table 12 lists the contour results obtained for the circular contour simulation. It can be concluded that the PDC with CCC provided an improvement in the mean contour error by as much as 0.0003m over both the TDC controllers and approximately 0.0001m over the PDC without CCC. In terms of standard of deviation of the contour error, the PDC with CCC provided an improvement of nearly 70% over the TDC controllers.

**Table 12: Magnitude of contour error mean, standard of deviation, and maximum**

	Mean(m)	S.D(m)	Max(m)
<b>TDC</b>	0.000470	0.000232	0.000887
<b>TDC with CCC</b>	0.000524	0.000248	0.000928
<b>PDC</b>	0.000216	0.000103	0.000329
<b>PDC with CCC</b>	0.000115	0.000061	0.000202

Comparison between the linear and circular contouring results indicates that the PDC with CCC significantly reduces the mean of the contouring error of the linear contours by as much as 60% over the TDC controllers. For the circular contour, the result achieved was higher at 75%. In all cases, both of the PDC controllers outperformed the equivalent TDC controllers with improved axial tracking and overall contour tracking performance. In cases where the tracking errors or contour errors of the PDC was comparable to the TDC controllers, the standard of deviation for these errors with the PDC controllers was significantly lower, therefore the PDC still provided better overall contour tracking.

## 4.6 OTHER SIMULATIONS

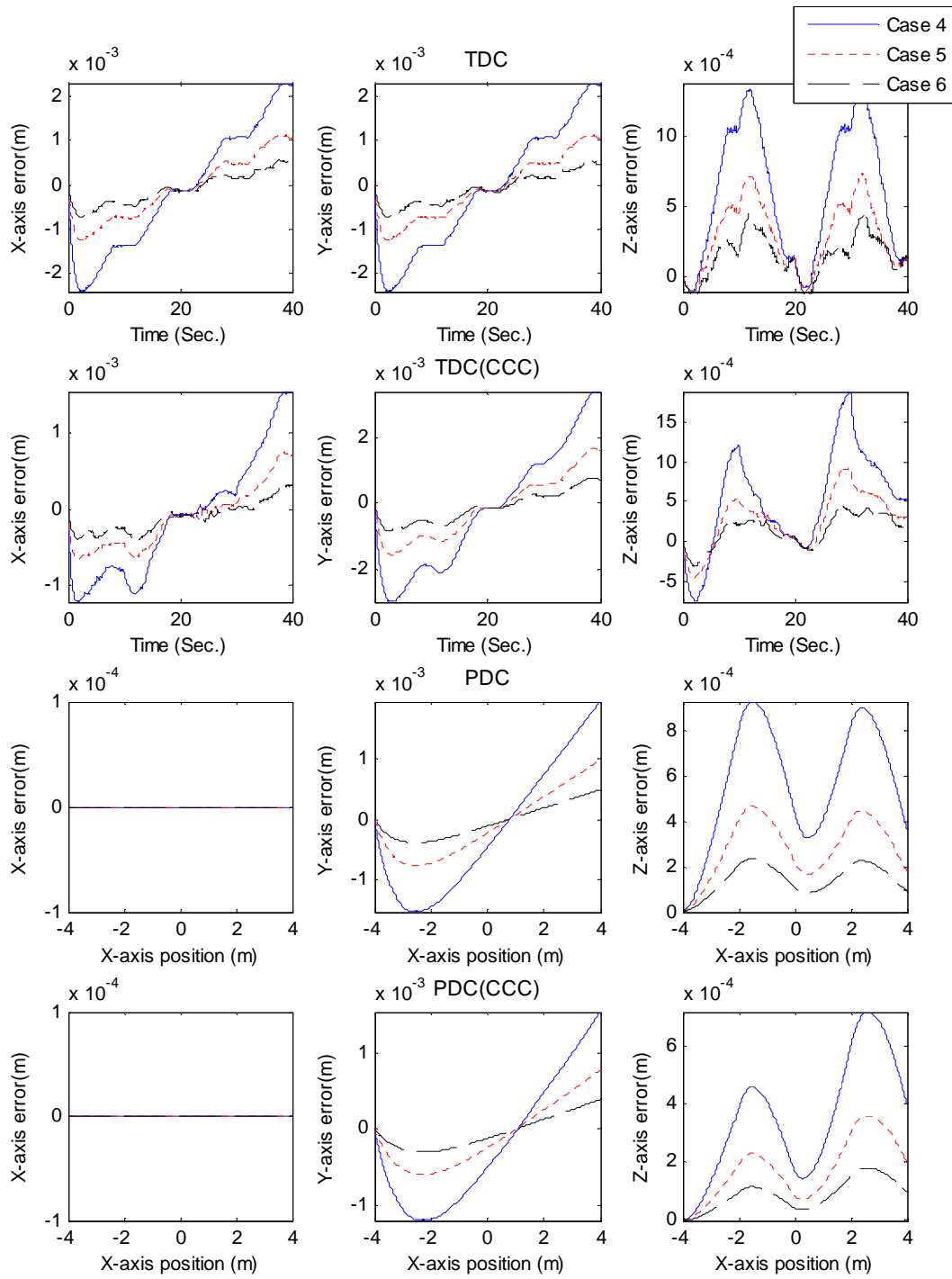
In order to further demonstrate the performance and robustness of the proposed position domain controller, various other simulations were conducted. These simulations varied from the initial by simulation by incorporating higher controller gains, higher CCC gains, and variation in the equidistant sampling of the PDC controller.

### 4.6.1 HIGH CONTROLLER GAINS

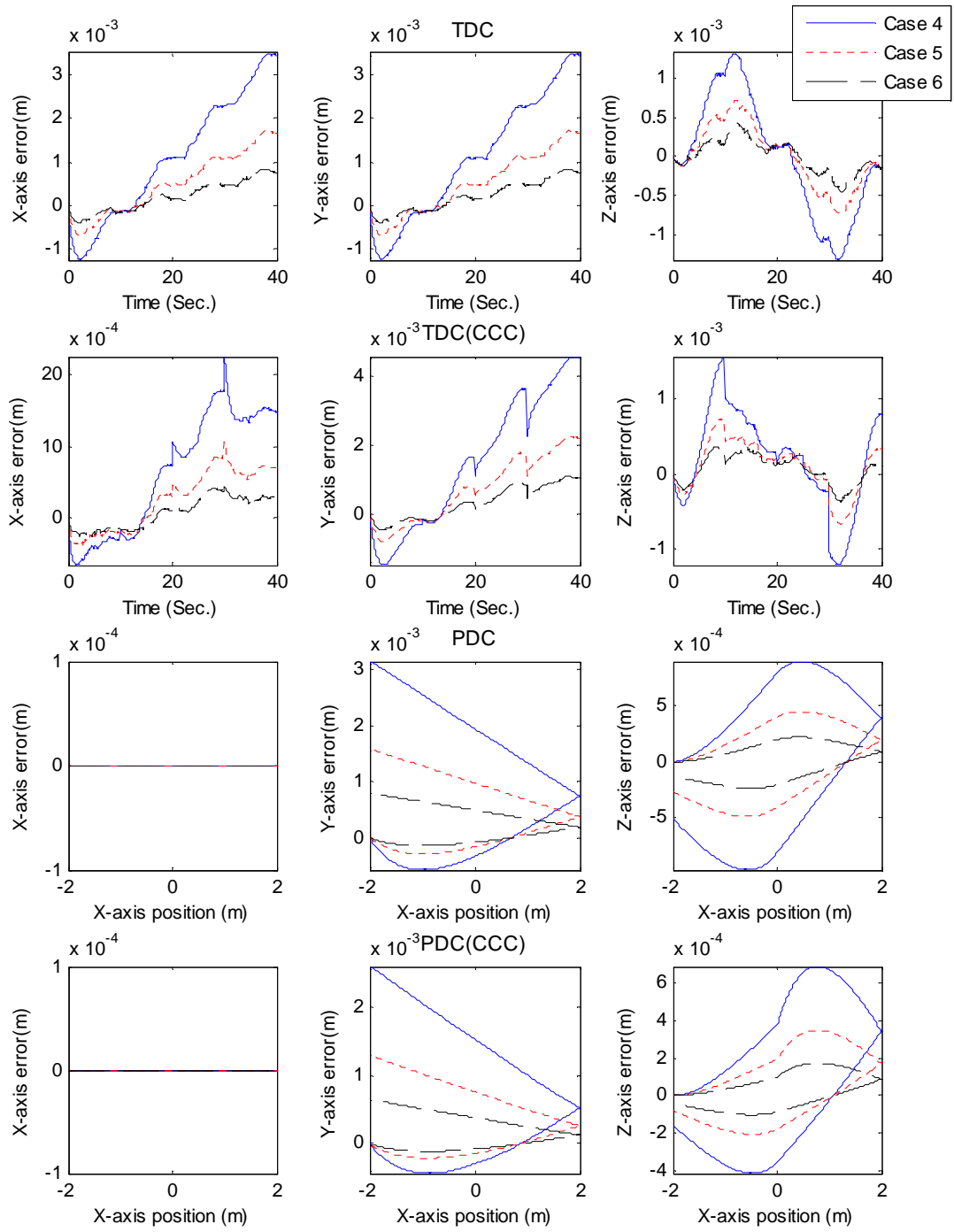
The higher controller gains used for additional simulations are listed in Table 13. These gains are based of controller gains from **Case 4**, which have been successively increased by a factor of 2 in **Case 5** and **Case 6** respectively. The axial tracking results for all contours are displayed from Figure 4-19 to Figure 4-21. From these plots, it is observed that the performance of all controllers is stable at the higher level of controller gains. Furthermore, both PDC controllers provided superior tracking performance over both the TDC controllers with the best overall performance from the PDC with CCC. The contouring results for all contours with higher gains are shown in Figure 4-22 and listed in Table 14. It is evident that for both linear contours, the utilization of higher controller gains resulted in improved contour tracking with the PDC with CCC controller providing the best contouring performance in all cases. For the circular contour, a similar improvement in the contour tracking was also observed. By doubling the controller gains, the contour errors for all controllers were reduced by approximately half.

**Table 13: Higher control gains**

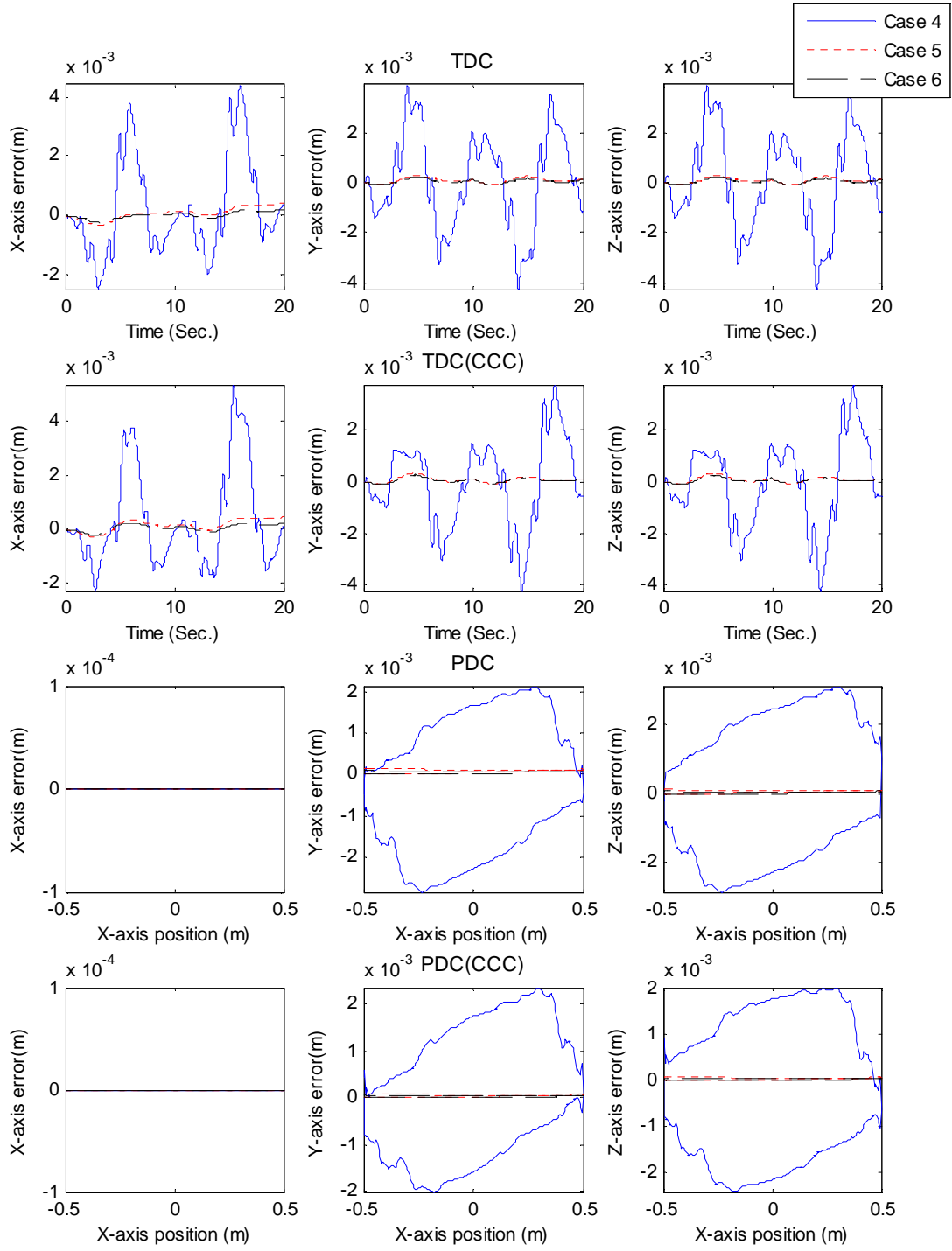
	<b>P</b>	<b>I</b>	<b>D</b>	<b>P<sub>ccc</sub></b>	<b>D<sub>ccc</sub></b>
<b>Case 5</b>	160000	100000	140000	160000	140000
<b>Case 6</b>	320000	200000	280000	320000	280000



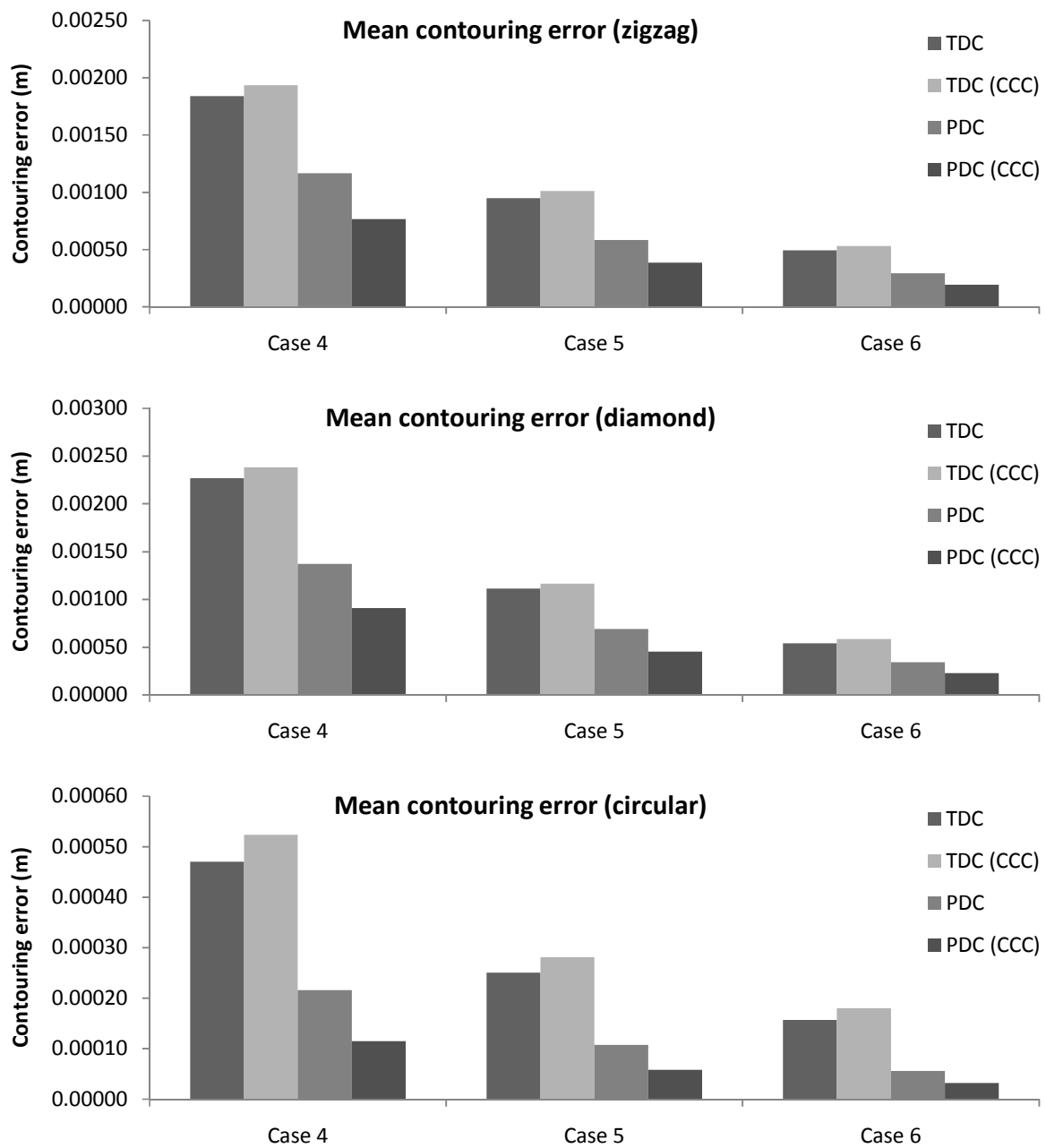
**Figure 4-19: Zigzag contour axial tracking errors with increased controller gains**



**Figure 4-20: Diamond contour axial tracking errors with increased controller gains**



**Figure 4-21: Circular contour axial tracking errors with increased controller gains**

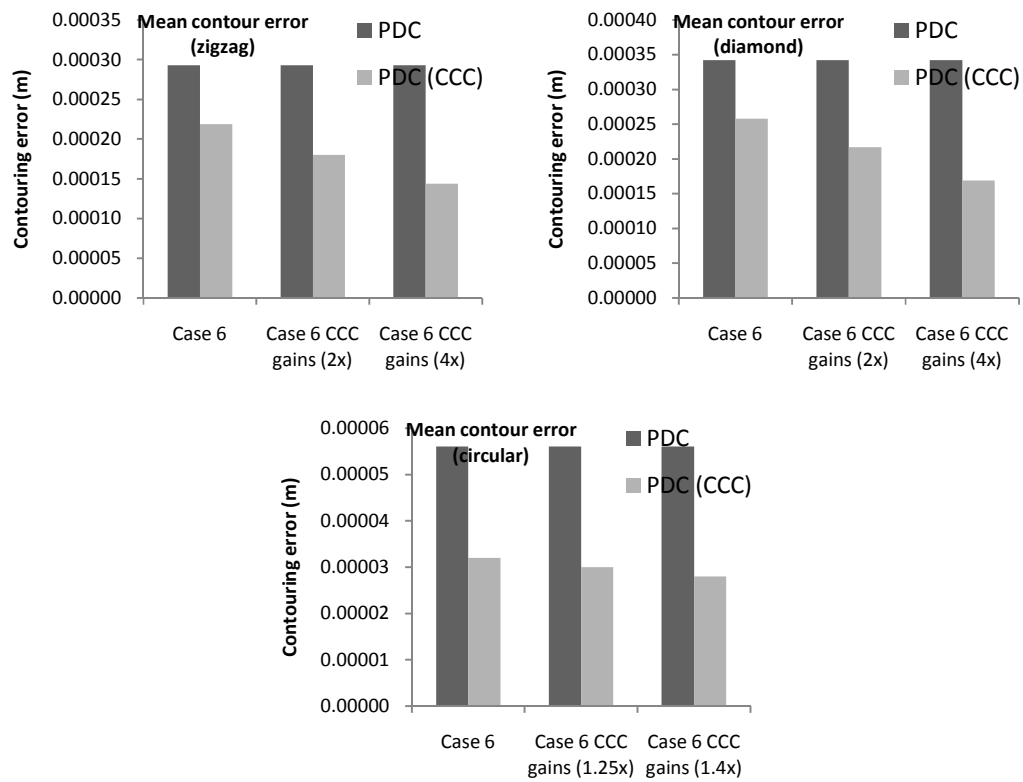


**Figure 4-22: Mean contouring error with increasing controller gains**

**Table 14: Contour errors with increased controller gains**

	<b>Case 4</b>	<b>Case 5</b>	<b>Case 6</b>
	Contour error(m)	Contour error(m)	Contour error (m)
<b>Zigzag</b>			
<b>TDC</b>	0.00184	0.00095	0.00049
<b>TDC (CCC)</b>	0.00194	0.00101	0.00053
<b>PDC</b>	0.00117	0.00058	0.00029
<b>PDC (CCC)</b>	0.00077	0.00038	0.00019
<b>Diamond</b>			
<b>TDC</b>	0.00227	0.00111	0.00054
<b>TDC (CCC)</b>	0.00239	0.00117	0.00059
<b>PDC</b>	0.00137	0.00069	0.00034
<b>PDC (CCC)</b>	0.00091	0.00045	0.00023
<b>Circular</b>			
<b>TDC</b>	0.00047	0.00025	0.00016
<b>TDC (CCC)</b>	0.00052	0.00028	0.00018
<b>PDC</b>	0.00022	0.00011	0.00006
<b>PDC (CCC)</b>	0.00012	0.00006	0.00003

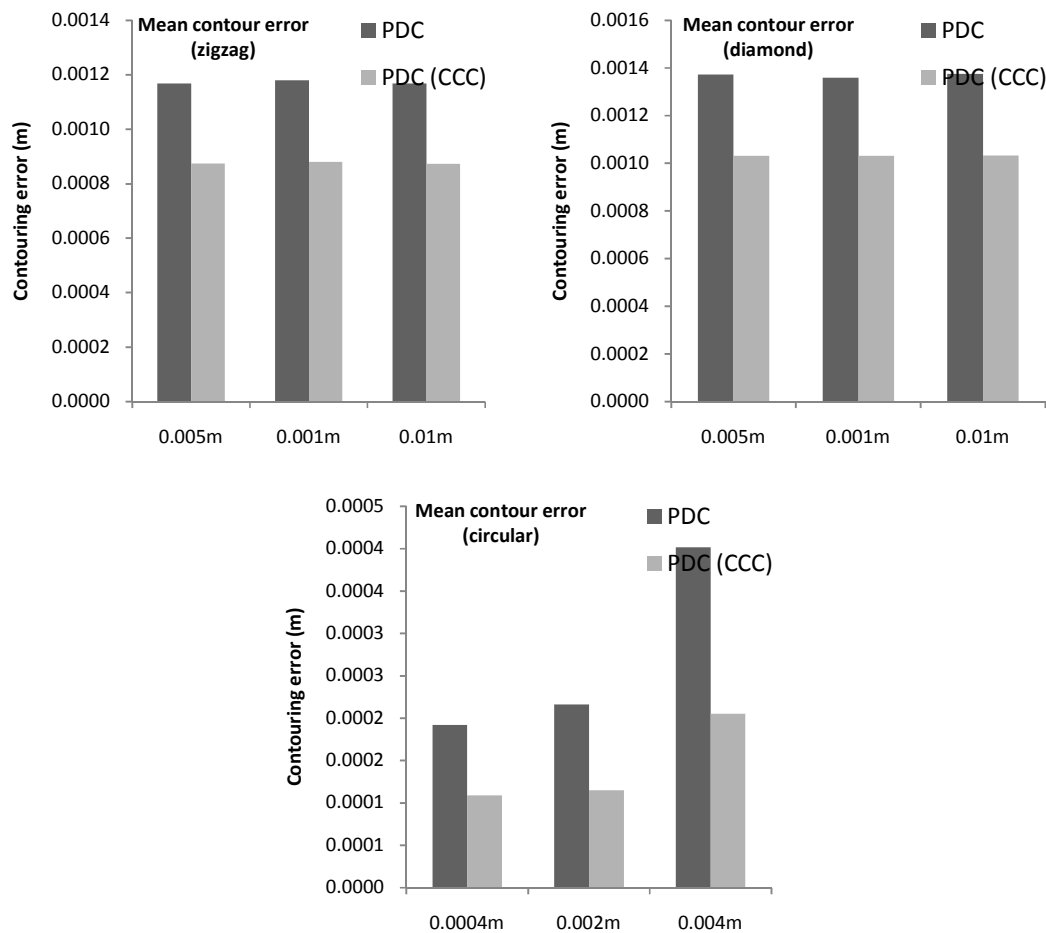
Simulations of higher CCC controller gains were also conducted which were based of **Case 6** controller gains. The only difference is that the CCC gains were increased proportional to the PID controller gains (by factors indicated in Figure 4-23 below). From this figure, the increased in the CCC gains resulted in improvements in the mean contour error of all three contours. The results show that there is a linear relationship between the increased in the CCC gains and the reduction in the mean contour error. Theoretically if the CCC gains were increased even to a higher value, the contour error achieved by the PDC with CCC would be even smaller.



**Figure 4-23: Contouring performance of PDC with increasing CCC gains**

#### 4.6.2 PDC EQUIDISTANT SAMPLING

One of the important factors that influence the overall performance of the PDC controller is the equidistant sampling of the reference axis. In the initial simulations, equidistant sampling of 0.005m and 0.002m were utilized for the linear and circular contours respectively. This section will demonstrate how the performance of the PDC controllers is influenced by the variation in the PDC sampling distance. The results of this simulation are shown in Figure 4-24, where the initial PDC sampling distance have decreased by 80% and increased by 100%.



**Figure 4-24: Contouring performance of PDC with varying sampling distance**

It is evident that varying the PDC sampling distance has no significant influence for the linear contours as the mean contouring errors of both the PDC and PDC with CCC remains relatively unchanged with both the decreased and increased in the PDC sampling distance. For the circular contour, there is a significant increase in the mean contour error (by a factor of approximately 2) when the sampling distance is increased by 100%. The opposite is observed when the sampling distance is decreased by 80% resulting in a slightly lower mean contour error. This is intuitive since the circular contour is nonlinear and the result of decreasing the resolution of the controller (by increasing the sampling distance) degraded its performance, while increasing the resolution (by decreasing the sampling distance) resulted in a slightly improved performance in terms of lower mean contour error. This means that equidistant sampling is an important factor for PDC, particularly with nonlinear contours, where there are higher variations in the position and velocity as well as acceleration of the axial trajectories.

## 4.7 MISMATCHING DYNAMICS

To further demonstrate the effectiveness of the PDC controller, the dynamics of the CNC system shown in Figure 3-1 were altered so that the axial dynamics are no longer matching. The purpose of this would be to enhance the simulation by creating a more realistic scenario. The new mismatching axial dynamics are listed in Table 15 below. Using Case 4 controller gains from the previous simulations, simulations with mismatching axial dynamics for all contour types indicate improved performance of the PDC over the TDC controller. With the addition of CCC, the improved performance of the PDC ranged from 50% to 70% lower tracking and contouring errors over the TDC controller. The results for the contouring performance for all contour types are listed in

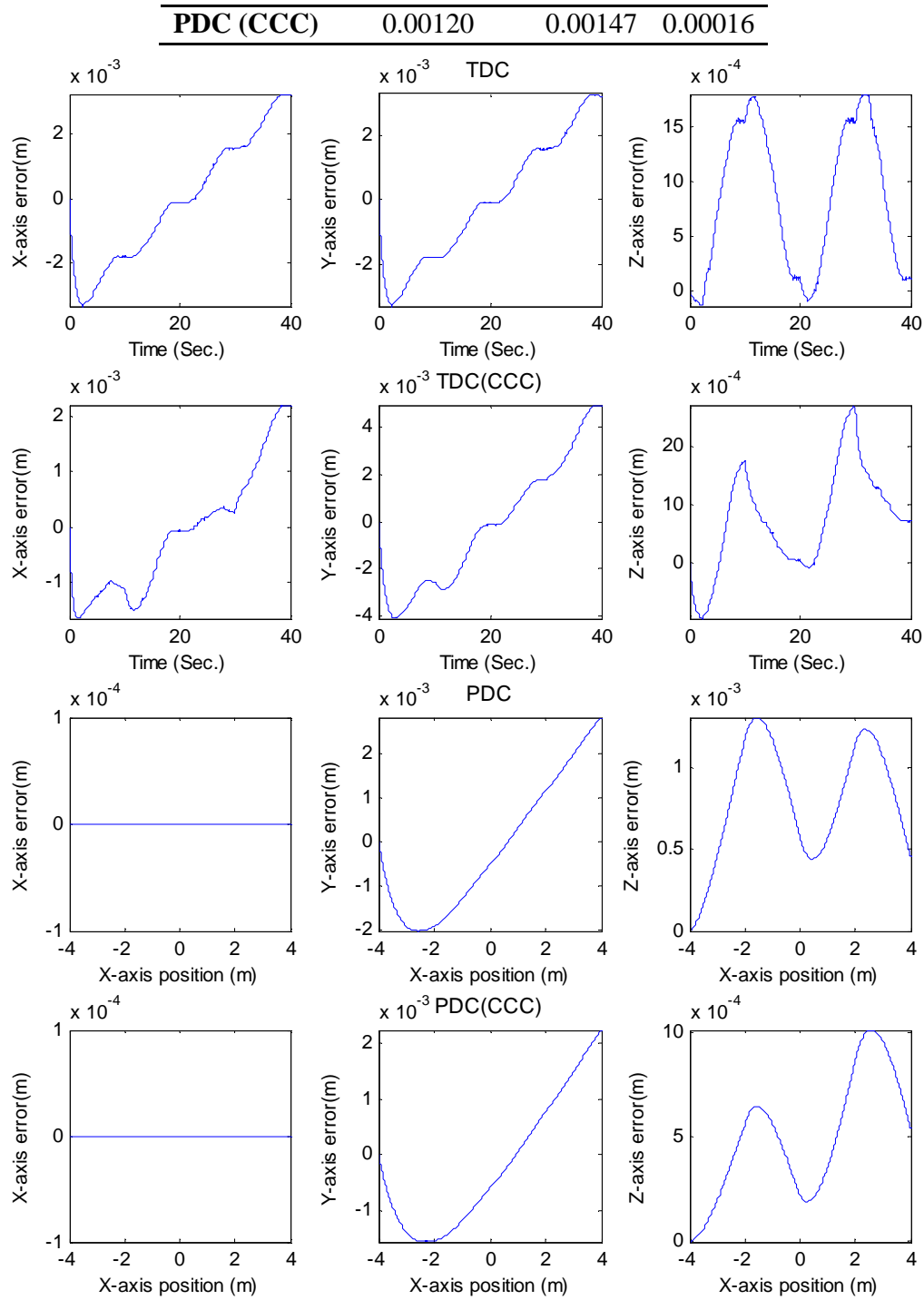
Table 16. The plots for the axial tracking errors are shown in Figure 4-25 to Figure 4-27. With mismatching axial dynamics, the PDC was able to provide better results over the TDC comparable with the non-mismatching case.

**Table 15: Mismatching system dynamic parameters**

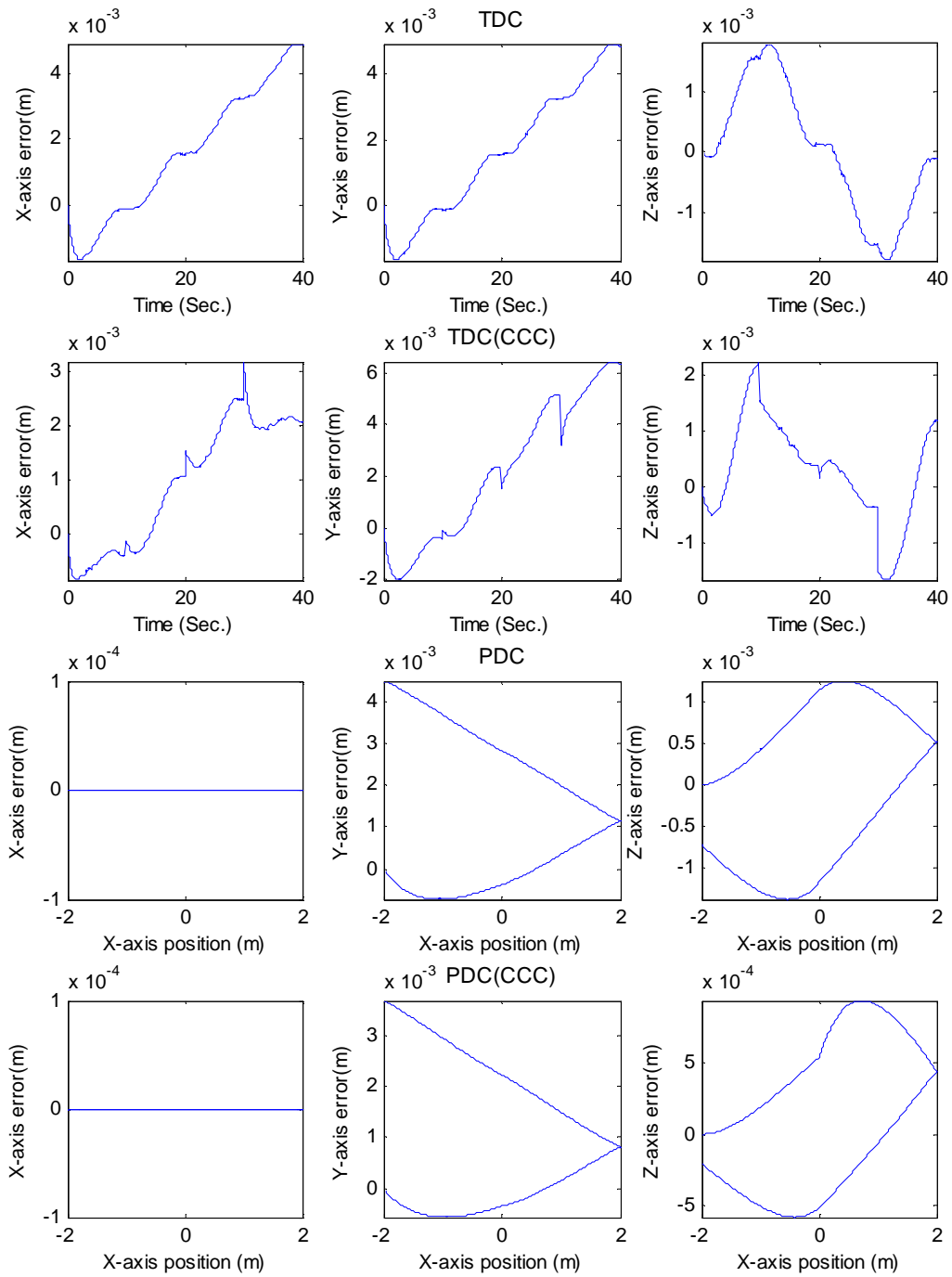
Axis	X	Y	Z
Mass ( $m_i$ [kg])	30	20	10
Damping ( $c_i$ [Ns/m])	20	40	20
Spring constant ( $k_i$ [N/m])	70	70	70

**Table 16: Mean contour error with mismatching dynamics**

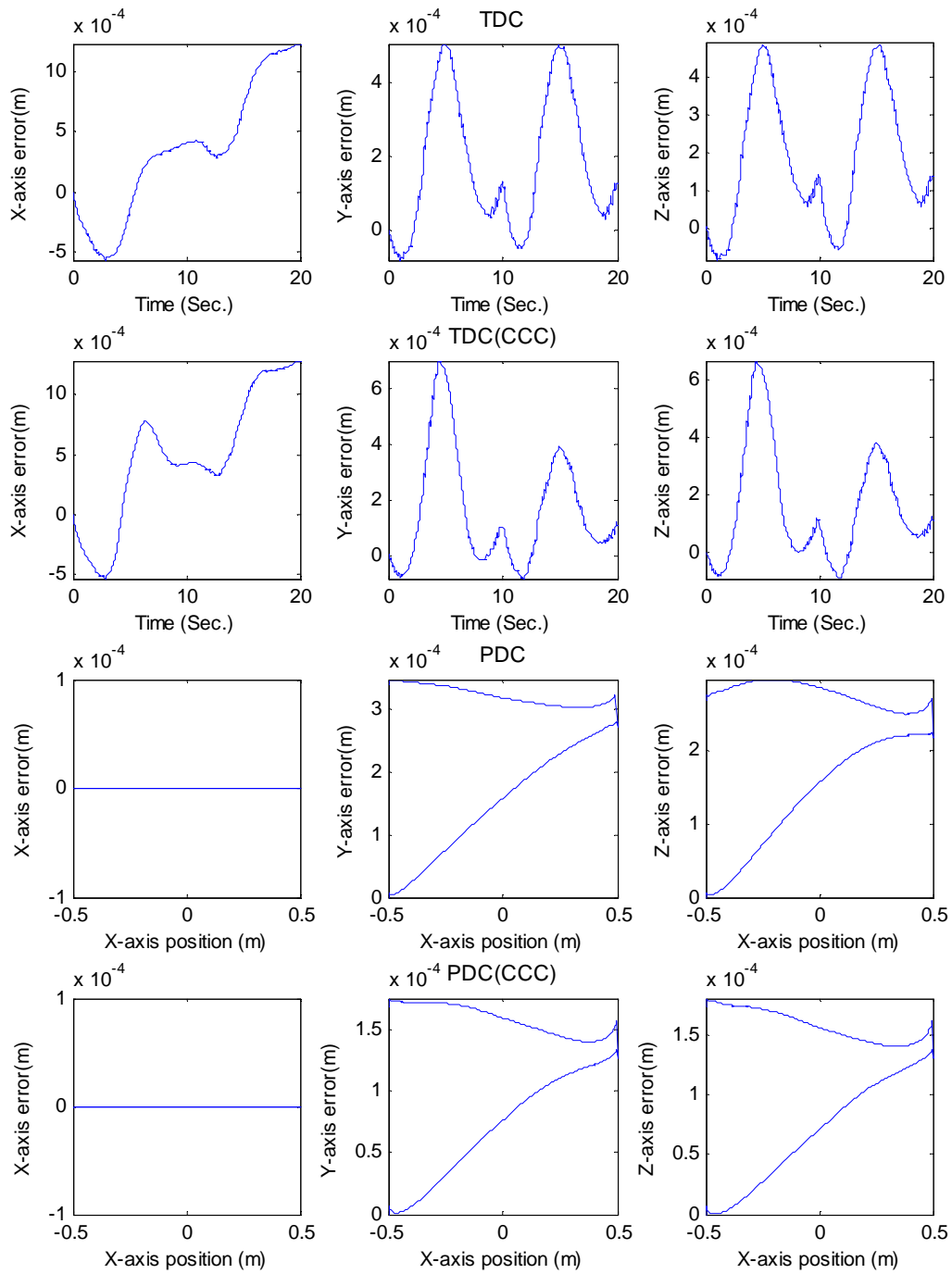
Mean contour error (m)			
	zigzag	diamond	circular
TDC	0.00257	0.00321	0.00065
TDC (CCC)	0.00271	0.00338	0.00073
PDC	0.00162	0.00197	0.00031



**Figure 4-25: Zigzag contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC**

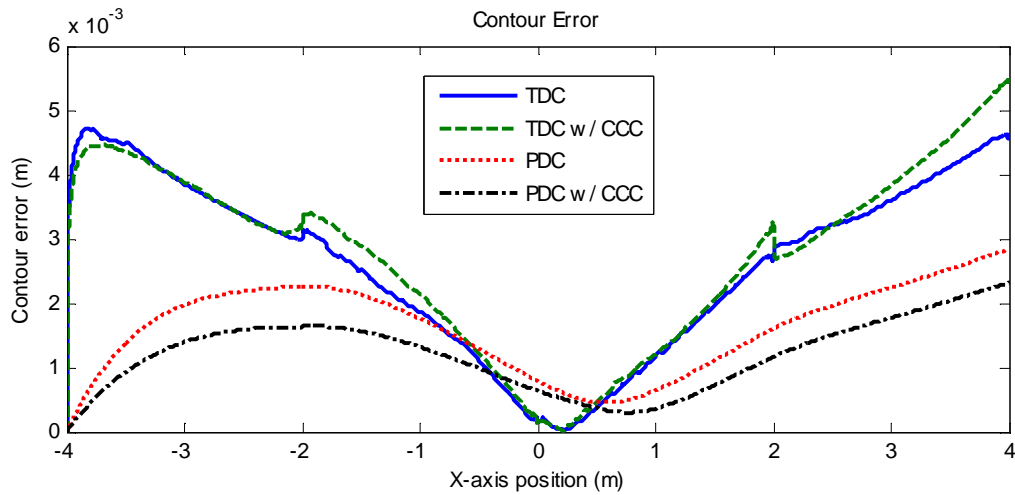


**Figure 4-26: Diamond contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC**

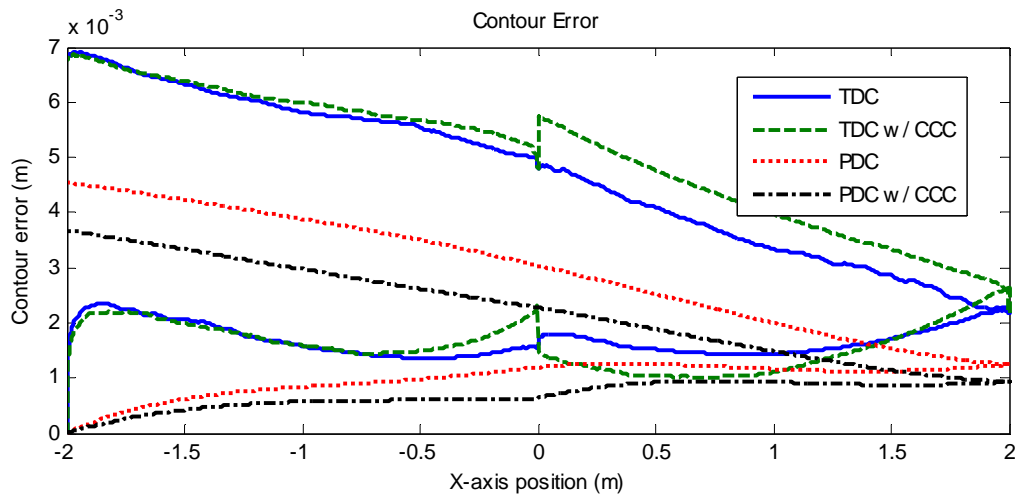


**Figure 4-27: Circular contour tracking error for TDC, TDC with CCC, PDC, and PDC with CCC**

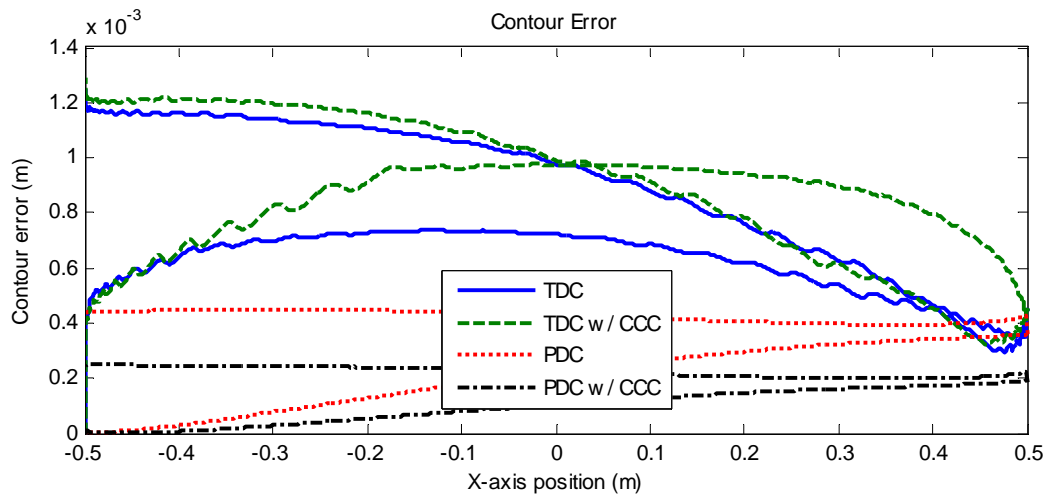
The contouring performance results for all three contours are displayed in Figure 4-28. It is clear that with mismatching axial dynamics, the PDC controller provided improved contouring performance over the TDC controllers for both linear and circular contours similarly to the non-mismatching case.



(a)



(b)



(c)

**Figure 4-28: Contouring error for (a) zigzag (b) diamond (c) circular contours with mismatching dynamics**

## 4.8 REMARKS

Simulation results indicate overall improvement in the tracking and contouring performances of the proposed PDC controller. For the linear contours, the PDC provided improvement in the mean contour error ranging from 35% to 40% over the TDC controllers. Improvement in the standard of deviation of the contour error was also observed to be in the range of 30% to 50% lower. With the addition of CCC to the PDC controller, further improvement in the mean contour error was achieved by approximately 60% over the TDC, while the standard of deviation of the contour error improved by as much as 70%. Similar results were also achieved for the circular contour simulation. An improvement of 55% in the mean contour error was observed for the PDC over the TDC in the circular contour, while the PDC with CCC achieved an improvement of 75% over the TDC. In terms of standard of deviation for the contour error, a 55% improvement was observed for the PDC and a 75% improvement was observed for the PDC with CCC over the TDC controller.

Simulation with higher control gains provided improved results for all controllers with the PDC with CCC providing the best tracking and contouring performance. It is observed that the reduction in the tracking errors and contouring errors for all controllers is approximately proportional to the increased in the controller gains. Simulation of higher CCC resulted in a similar observation. With the higher CCC gains, the PDC with CCC provided reduction in tracking and contouring errors proportional to the increased of the CCC gains.

Simulation with varying PDC sampling distance shows that there is no appreciable change in the tracking or contouring performance of the linear contour when the PDC sampling distance is increased by 100% or decreased by 80%. For the same change in the PDC sampling distance, there is a significant change in the performance of the controller with the circular

contour. This is the case since the circular contour is nonlinear and there is a high variation in the dynamics, particularly in the position of the contour. Therefore when the PDC sampling distance is increased, the resolution of the controller decreases, thus degrading the performance. The opposite is applied when the PDC sampling is decreased.

## Chapter 5 : CONCLUSION AND DISCUSSION

### 5.1 GENERAL REVIEW

In this thesis, a position domain controller was proposed and compared with an equivalent time domain controller for a 3DOF CNC machine. This was done by transforming the dynamics of the CNC system from time domain to position domain through a one-to-one mapping. In position domain, one of the axes in the system is taken as the reference and sampled equidistantly; therefore the reference axis yields zero tracking error. Only the tracking errors of the slave axes will contribute to the overall contouring error of the motion. The stability of the proposed position domain controller was demonstrated with the Lyapunov method. In addition to comparing time and position domain controllers, a cross-coupled control component was added to each of the controllers and comparative analysis conducted for all the controllers.

Linear and nonlinear contour type simulations were conducted for this system. The results indicate improved performance was achieved by the PDC controller over the equivalent TDC controller. This translate to lower tracking errors for the slave axes of  $y$  and  $z$ . In cases where the mean value of the tracking errors were comparable between the TDC and PDC controllers, the performance was still improved as the values of the tracking error standard deviation was lower with the PDC controller. In terms of contouring error performance, the PDC provided improved contour tracking over the TDC controller with lower mean contour error and standard of deviation. With the addition of CCC to the PDC controller, further improvement was achieved for both tracking and contouring performance for both types of contour.

## 5.2 MAIN CONTRIBUTIONS

The main contribution of this work is the development a of position domain controller for simple linear and nonlinear contours in CNC control. The research work results several published papers which includes:

- T. Dam and P. R. Ouyang, Position Domain Contour Tracking with Cross-coupled Control, Accepted by the 21st International Symposium on Industrial Electronics, Feb. 2012.
- P. R. Ouyang, V. Pano, and T. Dam, PID Contour Tracking Control in Position Domain, Accepted by the 21st International Symposium on Industrial Electronics, Feb. 2012.
- P. R. Ouyang, T. Dam, J. Huang, and W.J. Zhang, Contour Tracking Control in Position Domain, Accepted by Mechatronics, Dec., 2011.
- P.R. Ouyang and T. Dam, Position Domain PD Control: Stability and Comparison, The 2011 IEEE International Conference on Information and Automation (ICIA 2011), 8-13, 2011.
- T. Dam and P.R. Ouyang, Contour Control in Position Domain for CNC Machines, The 2011 IEEE International Conference on Information and Automation (ICIA 2011), 14-19, 2011. The Best Paper Award.
- P.R. Ouyang and T. Dam, Position domain PD control for contour tracking, Proceedings of the ASME 2010 International Mechanical Engineering Congress & Exposition (IMECE 2010).

### 5.3 **FUTURE DEVELOPMENT**

The work presented shows a position domain controller to be more effective at reducing tracking and contouring error for linear and circular contour type over an equivalent time domain controller. Further work is required to show the effectiveness of the position domain controller on arbitrary contours such as splines and for other geometric features like corners. The goal would be to demonstrate the potential of the position domain control to be used as an alternative to the time domain control.

# APPENDIX A:

## A.1 PRIMARY PROGRAM

```
%GLOBAL PARAMETERS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global kpx kpy kpz kix kiy kiz kdx kdy kdz kpc kdc kx ky kz cx cy cz ...
    f_pox f_poy f_poz xpos xi xf bz dt t tnum error errorx errory errorz ...
    xd dxd yd dyd zd dzd casel motion
%GENERAL PARAMETERS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%plot parameters
f_size=10;
m_size=4.5;
l_size=1.5;

%total motion time
T=10;Tx=T;Ty=T;Tz=T;

%sampling time
dt=0.001;t=0:dt:T;tnum=length(t);
t1=t;

%sampling position
delta_x=0.005;

%ode tolerance
tol=1e-5;
toll=1e-6;
deltal=1e-15;

%true/desired position, velocity, and acceleration
[r dr ddr]=f_getr(tnum,t,T);

%CCC gains factor
cccf=1;

%SYSTEM DYNAMICS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%dynamic parameters
m=1;k=50;c=7;
mx=m;my=m;mz=m;
kx=k;ky=k;kz=k;
cx=c;cy=c;cz=c;

%control gains
% kp=80000;ki=50000;kd=70000;casel=4;

kpx=kp;kpy=kp;kpz=kp;
kix=ki;kiy=ki;kiz=ki;
kdx=kd;kdy=kd;kdz=kd;
kpc=cccf*kp;kdc=cccf*kd;

%SECTOR 1%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%initial and final position
xi=[-4 -4 0];
```

```

xf=[-2 -2 2];
[f_pox f_poy f_poz f_po]=f_getf_po(xi,xf);
bz=xf(3)-(f_poz/f_pox)*xf(1);

%initial state
x0=[xi(1);0;xi(2);0;xi(3);0];

[xd dxd yd dyd zd dzd]=f_get_xyzr_linear(xi,f_po,r,dr);

errorx=zeros(tnum,1);
errory=zeros(tnum,1);
errorz=zeros(tnum,1);
options=odeset('RelTol',tol,'AbsTol',[tol tol tol tol tol tol]);
[tt,x]=ode23tb('xyz_t_control',t,x0,options);

errorx=zeros(tnum,1);
errory=zeros(tnum,1);
errorz=zeros(tnum,1);
options=odeset('RelTol',tol,'AbsTol',[tol tol tol tol tol tol]);
[tt,xc]=ode23tb('xyz_tc_control',t,x0,options);

%find the real velocity and acceleration of the x-axis
nj=abs(f_pox/delta_x+1);
xpos=zeros(3,1,nj);
xpos(:, :, 1)=[xi(1);0;0];

for j=2:nj
    xpos(1,1,j)=xi(1)+(j-1)*delta_x;
    for i=1:tnum
        if x(i,1)>=xpos(1,1,j); break
        end
    end
    xpos(2,1,j)=x(i,2);
    if i>1
        xpos(3,1,j)=(x(i,2)-x(i-1,2))/dt;
    end
end
xpos0=xpos(:, :, nj);

tt=squeeze(xpos(1,1,:));
tt(1)=tt(1)+ delta1;
tt(nj)=tt(nj)- delta1;
nt=size(xpos(1,1,:),3);
y0=[xi(2);0];
z0=[xi(3);0];
yz0=[y0;z0];

[yx zx yzx]=f_get_PDC_linear(nt,tt,y0,z0,yz0,tol1);

%TDC reference
[xr1 xdr1 yr1 ydr1 zr1 zdr1]=f_get_xyzr_linear(xi,f_po,r,dr);
%TDC actual
[xa1 xda1 ya1 yda1 za1 zda1]=f_get_xyza(x);
%TDC w/ CCC actual
[xac1 xdac1 yac1 ydac1 zac1 zdac1]=f_get_xyza(xc);
%PDC reference
[xrp1 yrp1 zrp1]=f_get_xyzrp_linear(xpos,f_po,bz);

```

```

%PDC actual
[xap1 yap1 zap1]=f_get_xyzap(xpos,yx,zx);
%PDC w/ CCC actual
[xapc1 yapc1 zapc1]=f_get_xyzapc(xpos,yzx);
%calculate tracking error
e_tl=f_te(xr1,yr1,zr1,xal,yal,zal);
e_tcl=f_te(xr1,yr1,zr1,xac1,yac1,zac1);
e_pl=f_te(xrpl,yrpl,zrpl,xap1,yap1,zap1);
e_pcl=f_te(xrpl,yrpl,zrpl,xapc1,yapc1,zapc1);

%SECTOR 2%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

## A.2 SECONDARY FUNCTIONS

```
function dx=xyz_t_control(tt,x)
global kpx kpy kpz kix kiy kiz kdx kdy kdz kx ky kz cx cy cz dt t tnum
global xd dxd yd dyd zd dzd

dx=zeros(6,1);

for i=1:tnum
    if t(i)>=tt break
    end
end

errorx(i)=xd(i)-x(1);
errorx(i)=yd(i)-x(3);
errorz(i)=zd(i)-x(5);

dx(1)=x(2);
dx(2)=-cx*x(2)-kx*x(1)+kpx*(xd(i)-x(1))+kdx*(dxd(i)-x(2))+kix*sum(errorx)*dt;

dx(3)=x(4);
dx(4)=-cy*x(4)-ky*x(3)+kpy*(yd(i)-x(3))+kdy*(dyd(i)-x(4))+kiy*sum(errory)*dt;

dx(5)=x(6);
dx(6)=-cz*x(6)-kz*x(5)+kpz*(zd(i)-x(5))+kdz*(dzd(i)-x(6))+kiz*sum(errorz)*dt;

function dx=xyz_tc_control(tt,x)
global kpx kpy kpz kix kiy kiz kdx kdy kdz kx ky kz cx cy cz dt t tnum
global xd dxd yd dyd zd dzd f_pox f_poy f_poz kpc kdc

dx=zeros(6,1);

for i=1:tnum
    if t(i)>=tt
        break;
    end
end

errorx(i)=xd(i)-x(1);
errorx(i)=yd(i)-x(3);
errorz(i)=zd(i)-x(5);

angley=atan(f_poy/f_pox)*180/pi;
anglez=atan(f_poz/f_pox)*180/pi;

ex=errorx(i);
dex=dxd(i)-x(2);
ey=errorx(i);
dey=dyd(i)-x(4);

ccx=sin(angley);
dcx=cos(angley);
ccy=cos(angley);
dcy=-sin(angley);
```

```

ecy=-ccx*ex+ccx*ex;
decy=-ccx*dex+ccy*dey-dcx*ex+dcy*ey;

ez=errorz(i);
dez=dzd(i)-x(6);

ccx2=sin(anglez);
dcx2=cos(anglez);
ccz=cos(anglez);
dcz=-sin(anglez);

ecz=-ccx2*ex+ccx2*ex;
decz=-ccx2*dex+ccz*dez-dcx2*ex+dcz*ez;

dx(1)=x(2);
dx(2)=(-cx*x(2)-kx*x(1)+kpx*(xd(i)-x(1))+kdx*(dxd(i)-
x(2))+kix*sum(errorx)*dt)-ccx*(kpc*ecy+kdc*decy)-ccx2*(kpc*ecz+kdc*decz);

dx(3)=x(4);
dx(4)=(-cy*x(4)-ky*x(3)+kpy*(yd(i)-x(3))+kdy*(dyd(i)-
x(4))+kiy*sum(errory)*dt)+ccy*(kpc*ecy+kdc*decy);

dx(5)=x(6);
dx(6)=(-cz*x(6)-kz*x(5)+kpz*(zd(i)-x(5))+kdz*(dzd(i)-
x(6))+kiz*sum(errorz)*dt)+ccz*(kpc*ecz+kdc*decz);

function dyx=y_axis_p_control(t,y)
global kpy kiy kdy ky cy f_pox f_poy xpos dt error

dyx=zeros(2,1);

ydx=f_poy/f_pox*t;
dydx=f_poy/f_pox;

nt=size(xpos(1,1,:),3);
for i=1:nt
    if xpos(1,1,i)>=t
        break;
    end
end
error(i)=ydx-y(1);

vx=xpos(2,1,i);
p=(-(xpos(3,1,i)+cy*xpos(2,1,i)+kdy)*y(2)-
(ky+kpy)*y(1)+kpy*ydx+kdy*dydx+kiy*sum(error)*dt);

if vx~=0
    dyx(2)=p/vx^2;
end
dyx(1)=y(2);

function dzx=z_axis_p_control(t,z)
global kpz kiz kdz kz cz f_pox f_poz xpos dt error bz
d zx=zeros(2,1);

```

```

zdx=f_poz/f_pox*t+bz;
dzdx=f_poz/f_pox;

nt=size(xpos(1,1,:),3);
for i=1:nt
    if xpos(1,1,i)>=t
        break;
    end
end
error(i)=zdx-z(1);

vx=xpos(2,1,i);
p=(-(xpos(3,1,i)+cz*xpos(2,1,i)+kdz)*z(2)-
(kz+kpz)*z(1)+kpz*zdx+kdz*dzdx+kiz*sum(error)*dt);

if vx~=0
    dzx(2)=p/vx^2;
end
dzx(1)=z(2);

function dyzx=yz_axis_p_control(t,yz)
global kpy kiy kdy ky cy f_pox f_poy xpos dt errory errorz
global kpz kiz kdz kz cz f_poz angle kpc kdc bz

dyzx=zeros(4,1);

ydx=f_poy/f_pox*t;
dydx=f_poy/f_pox;
zdx=f_poz/f_pox*t+bz;
dzdx=f_poz/f_pox;

angle=atan(f_poz/f_poy)*180/pi;

nt=size(xpos(1,1,:),3);
for i=1:nt
    if xpos(1,1,i)>=t
        break;
    end
end
errory(i)=ydx-yz(1);
errorz(i)=zdx-yz(3);

ey=errory(i);
dey=dydx-yz(2);
ez=errorz(i);
dez=dzdx-yz(4);

ccy=sin(angle);
dcy=cos(angle);
ccz=cos(angle);
dcz=-sin(angle);

ec=-ccy*ey+ccz*ez;
dec=-ccy*dey+ccz*dez-dcy*ey+dcz*ez;

vx=xpos(2,1,i);

```

```

py=(-(xpos(3,1,i)+cy*xpos(2,1,i)+kdy)*yz(2)-
(ky+kpy)*yz(1)+kpy*ydx+kdy*dydx+kiy*sum(errory)*dt)-ccy*(kpc*ec+kdc*dec);

if vx~=0
    dyzx(2)=py/vx^2;
end
dyzx(1)=yz(2);

pz=(-(xpos(3,1,i)+cz*xpos(2,1,i)+kdz)*yz(4)-
(kz+kpz)*yz(3)+kpz*zdx+kdz*dzdx+kiz*sum(errorz)*dt)+ccz*(kpc*ec+kdc*dec);

if vx~=0
    dyzx(4)=pz/vx^2;
end
dyzx(3)=yz(4);

```

## BIBLIOGRAPHY

- Arzen, K. (1999). A Simple Event-based PID controller. *Proceeding of the 14th IFAC World Congress*.
- Astrom, K., & Bernhardsson, B. (1999). Comparison of Periodic and Event Based Sampling for First-order Stochastic Systems. *Proceedings of the 14th IFAC World Congress*.
- Astrom, K., Hagander, P., & Sternby, J. (1984). Zeroes of Sampled Systems., 20, pp. 31-38.
- Barton, K. L., & Alleyne, A. G. (2008). A cross-coupled iterative learning control design for precision motion control. *IEEE on Control Systems Technology* , 16 (4), 1218-1230.
- Bristow, D., Tharayil, M., & Alleyne, A. (2006, June). A survey of iterative learning control: a learning-based method for high-performance tracking control. *Control Systems, IEEE* , 26 (3), pp. 96-114.
- Cheng, Y., & Chin, J. (2003). Machining contour errors as ensembles of cutting, feeding, and machine structure effects. *International Journal of Machine Tools and Manufacture* , 43 (10), 1001-1014.
- Craig, & J, J. (1989). *Introduction to robotics mechanics and control*. Reading: Addison Wesley Longman.
- Erkorkmaz, K., Yeung, C., & Altintas, Y. (2006). Virtual CNC system. Part II. High speed contouring application. *International Journal of Machine Tools and Manufacture* , 46, 1124-1138.
- Fang, R. W., & Chen, J. (2002). Cross-coupling control for a direct-drive robot. *JSME International Journal Series C* , 45 (3), 749-757.
- Haack, B., & Tomizuka, M. (1991). The effect of adding zeroes to feedforward. *Journal of Dynamics Systems, Measurement and Conntrol* , 113 (1), 6-10.

- Hu, C., Yao, B., & Wang, Q. (2009). Coordinated contouring controller design for an industrial biaxial linear motor driven gantry. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, (pp. 1810-1815).
- Inoue, T., Nakano, M., Kubo, T., Matsumoto, S., & Baba, H. (1981). High accuracy control of a proton synchrotron magnet power supply. *Proceeding of the 8th world congress of IFAC*, (p. 216).
- Khalil, H. (2002). *Nonlinear Systems* (3rd Edition ed.). Prentice Hall.
- Kim, D., & S., K. (1996). An iterative learning control method with application for CNC machine tools. *IEEE transactions on industry applications* , 32 (1), 66-72.
- Koren, Y. (1983). *Computer Control of Manufacturing Systems*. New York: McGraw-Hill.
- Koren, Y. (1997). Control of Machine Tools. *Journal of Manufacturing Science and Engineering* , 119, 749-755.
- Koren, Y. (1980). Cross-coupled biaxial computer control for manufacturing systems. *ASME Journal of Dynamics Systems, Measurement, and Control* , 102, 265-272.
- Koren, Y., & Lo, C.-C. (1991). Variable-Gain Cross-Coupling Controller for Contouring. *Annals of the CIRP* , 40 (1), 371-374.
- Longman, R. (2000). Iterative learning control and repetitive control for engineering practice. *International Journal of Control* , 73 (10), 930-954.
- Ogata, K. (1997). *Modern Control Engineering*. Upper Saddle River: Prentice-Hall.
- Ouyang, P., & Dam, T. (2010). Position Domain PD Control for Contour Tracking. *Proceedings of the ASME 2010 International Mechanical Engineering Congress & Exposition*.

- Ouyang, P., Zhang, W., & Wu, F. (2002). Nonlinear PD Control for Trajectory Tracking with Consideration of the Design for Control Methodology. *Proceedings of 2002 IEEE ICRA*, 4, pp. 4126-4161. Washington.
- Ramesh, R., Mannan, M., & Poo, A. (2005). Tracking and contour error control in CNC servo systems. *International Journal of Machine Tool and Manufacture Design, Research, and Application* , 45 (3), 301-326.
- Schinkel, M., Chen, W., & Rantzer, A. (2002). Optimal control for systems with varying sampling rate. *Proceedings of the American Control Conference*, 4, pp. 2979-2984.
- Shih, Y., Chen, C., & Lee, A. (2002). A novel cross-coupling control design for Bi-axis motion. *International Journal of Machine Tools and Manufacture* , 42 (14), 1539-1548.
- Srinivasan, K., & Kulkarni, P. (1990). Cross-coupled control of biaxial feed drive servomechanism. *Journal of Dynamics Systems, Measurement, and Control* , 112 (2), 225-232.
- Suh, S.-H., Kang, S.-K., Chung, D.-H., & Stroud, I. (2008). *Theory and Design of CNC Systems*. London: Springer.
- Tan, K., Chua, K., Zhao, S., Yang, S., & M.T, T. (2009). Repetitive control approach towards automatic tuning of Smith predictor controllers. *ISA Transactions* , 48 (1), 16-23.
- Tomizuka, M. (1987). Zero Phase Error Tracking Algorithm for Digital Control. *Journal of Dynamics Systems, Measurement, and Control* , 109 (1), 65-68.
- Tsao, T.-C., & Tomizuka, M. (1987). Adaptive zero phase error tracking algorithm for digital control. *Journal of Dynamic Systems, Measurement, and Control* , 109 (4), 349-354.
- Van Den Braembussche, P., Swevers, J., Van Brussel, H., & Vanherck, P. (1996). Accurate tracking control of linear synchronous motor machine tool axes. *Mechatronics* , 6 (5), 507-521.

Weck, M., & Ye, G. (1990). Sharp corner tracking using the IKF control strategy. *CIRP Annals - Manufacturing Technology* , 39 (1), 437-441.

Yeh, S.-S., & Hsu, P.-L. (2003). Analysis and Design of Integrated Control for Multi-Axis Motion Systems. *IEEE Transactions on Control Systems Technology* , 11 (3), 375-382.