

HYBRID PD SLIDING MODE CONTROL FOR ROBOTIC MANIPULATORS

by

John Michael Acob, B.Eng
Aerospace Engineering
Ryerson University, 2011

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, 2015

© John Michael Acob 2015

AUTHOR'S DECLARATION

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

Hybrid PD Sliding Mode Control for Robotic Manipulators

John Michael Acob, Master of Applied Science

Aerospace Engineering

Ryerson University, Toronto 2015

This thesis proposes a new control law for the purpose of providing improved tracking and contouring performance of robotic manipulators. The rationale behind the development of this controller involves the hybridization of existing proportional-derivative (PD) and sliding mode control (SMC) laws. The new control law retains similar ease of implementation as traditional PD/PID controllers with the added benefit of a nonlinear switching component inherent from sliding mode control systems. In addition, it eliminates the need for a priori knowledge of the system dynamics that are required in standard SMC laws. The stability analysis of the proposed control law is conducted through the Lyapunov method. Simulations using linear and nonlinear contours, and under varying dynamic conditions are performed in order to compare its performances to existing control schemes. The proposed hybrid PD-SMC control law is proven to provide good, robust tracking and contouring performance.

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincere gratitude to my supervisor, Dr. Puren Ouyang. Without his continued guidance, profound support, and extraordinary patience, the completion of this thesis would not have been possible.

I also wish to express my gratitude to Mr. Yu Lin and Mr. Vangjel Pano, who were always available for their advice, opinions, and a helping hand when it was needed most.

Finally, I would like to extend my heartfelt thanks to my family. Thank you all for your unbelievable love and support. You have my deepest gratitude for your kindness and generosity.

DEDICATION

*To my parents,
my wife, and my son*

TABLE OF CONTENTS

Author's Declaration	ii
Abstract	iii
Acknowledgements	iv
Dedication	v
Table of Contents	vi
List of Tables	x
List of Figures	xiv
Acronyms	xix
Nomenclature	xx
Chapter 1 : Introduction	1
1.1 Background	1
1.2 Motivation and Objectives	5
1.3 Organization of Contents	6
Chapter 2 : Literature Review	8
2.1 PD/PID Control	8
2.1.1 General PID Control	9
2.1.2 PD with Desired Gravity Compensation	10
2.1.3 Nonlinear PD/PID Control	11
2.2 Other Control Methods	13
2.2.1 Event-Driven Control	14
2.2.2 Adaptive Control	15
2.2.3 Learning Control	18

2.3	Hybridization of Control Laws	19
2.3.1	Iterative Learning Control.....	19
2.3.2	Switching Control	21
2.3.3	Sliding Mode Control.....	22
2.4	Tracking and Contour Error	24
2.4.1	Tracking and Contour Error	24
2.4.2	Contour Error Calculation.....	25
2.5	Remarks.....	26
Chapter 3 : Hybrid PD Sliding Mode Control		28
3.1	Dynamic Model.....	28
3.2	PD Control Formulation.....	29
3.3	SMC Formulation.....	30
3.4	Hybrid PD Sliding Mode Control Formulation.....	32
3.4.1	Principle for Hybridization.....	32
3.4.2	Formulation	33
3.4.3	Properties of the Dynamic Model	35
3.5	Stability Analysis	36
3.5.1	Notations and Theorem	36
3.5.2	Proposition	37
3.5.3	Proof of Stability	39
3.6	Remarks.....	41
Chapter 4 : Simulation and Results.....		43
4.1	Simulation Setup.....	43
4.1.1	Serial Manipulator.....	43
4.1.2	Parallel Manipulator	45

4.2	Trajectory Planning for Simulations	46
4.2.1	Linear Contours	47
4.2.2	Nonlinear Contours	56
4.2.3	Control Parameter Selection.....	65
4.3	Serial Manipulator Simulation Results	71
4.3.1	Zigzag Contour.....	72
4.3.2	Diamond Contour	77
4.3.3	Circular Contour.....	81
4.3.4	Elliptical Contour	86
4.4	Parallel Manipulator Simulation Results.....	90
4.4.1	Zigzag Contour.....	91
4.4.2	Diamond Contour	95
4.4.3	Circular Contour.....	99
4.4.4	Horizontal Elliptical Contour	103
4.4.5	Vertical Elliptical Contour	107
4.5	Other Simulations.....	111
4.5.1	High-Speed Movement Simulations	112
4.5.2	End-Effector Payload Simulations	114
4.5.3	High Control Gains	115
4.6	Remarks.....	119
Chapter 5 : Conclusions and Future Work.....		122
5.1	General Review	122
5.2	Main Contributions	123
5.3	Future Work	123
List of Appendices		125

Serial 3R Manipulator Kinematics and Dynamics.....	125
Forward Kinematics.....	125
Inverse Kinematics.....	125
Dynamics	126
Parallel 5R Manipulator Kinematics and Dynamics.....	127
Forward Kinematics.....	127
Inverse Kinematics.....	128
Dynamics	129
Bibliography	132

LIST OF TABLES

Table 4.1: Structural Parameters of the 3R Serial Manipulator.....	44
Table 4.2: Structural Parameters of the 5R Parallel Manipulator.....	46
Table 4.3: Linear Contour Segment End-effector Endpoints for Serial Manipulator.....	48
Table 4.4: Linear Contour Segment Joint Position Endpoints for Serial Manipulator.....	49
Table 4.5: Linear Contour Segment End-effector Endpoints for Parallel Manipulator.....	52
Table 4.6: Linear Contour Segment Joint Position Endpoints for Parallel Manipulator.....	52
Table 4.7: Circular Contour Parameters for Serial Manipulator.....	57
Table 4.8: Circular Contour Initial and Final Positions for Serial Manipulator.....	57
Table 4.9: Elliptical Contour Parameters for Serial Manipulator.....	59
Table 4.10: Elliptical Contour Initial and Final Positions for Serial Manipulator.....	59
Table 4.11: Circular Contour Parameters for Parallel Manipulator.....	61
Table 4.12: Circular Contour Initial and Final Positions for Parallel Manipulator.....	61
Table 4.13: Elliptical Contour Parameters for Parallel Manipulator.....	62
Table 4.14: Elliptical Contour Initial and Final Positions for Parallel Manipulator.....	63
Table 4.15: Control Parameter Variation Ranges.....	66
Table 4.16: Control Parameters for Serial Manipulator Simulations.....	72
Table 4.17: Mean and Standard Deviation of Axial Tracking Errors for Zigzag Motion of Serial Manipulator.....	73
Table 4.18: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Zigzag Motion of Serial Manipulator.....	74
Table 4.19: Magnitude of Contour Tracking Performance for Zigzag Motion of Serial Manipulator.....	75
Table 4.20: Maximum Input Torques for Zigzag Motion of Serial Manipulator.....	77
Table 4.21: Mean and Standard Deviation of Axial Tracking Errors for Diamond Motion of Serial Manipulator.....	78
Table 4.22: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Diamond Motion of Serial Manipulator.....	78
Table 4.23: Magnitude of Contour Tracking Performance for Diamond Motion of Serial Manipulator.....	79
Table 4.24: Maximum Input Torques for Diamond Motion of Serial Manipulator.....	81

Table 4.25: Mean and Standard Deviation of Axial Tracking Errors for Circular Motion of Serial Manipulator.....	82
Table 4.26: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Circular Motion of Serial Manipulator	83
Table 4.27: Magnitude of Contour Tracking Performance for Circular Motion of Serial Manipulator.....	84
Table 4.28: Maximum Input Torques for Circular Motion of Serial Manipulator	85
Table 4.29: Mean and Standard Deviation of Axial Tracking Errors for Elliptical Motion of Serial Manipulator	86
Table 4.30: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Elliptical Motion of Serial Manipulator.....	87
Table 4.31: Magnitude of Contour Tracking Performance for Elliptical Motion of Serial Manipulator.....	88
Table 4.32: Maximum Input Torques for Elliptical Motion of Serial Manipulator.....	90
Table 4.33: Control Parameters for Parallel Manipulator Simulations.....	91
Table 4.34: Mean and Standard Deviation of Axial Tracking Errors for Zigzag Motion of Parallel Manipulator.....	92
Table 4.35: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Zigzag Motion of Parallel Manipulator	92
Table 4.36: Magnitude of Contour Tracking Performance for Zigzag Motion of Parallel Manipulator.....	93
Table 4.37: Maximum Input Torques for Zigzag Motion of Parallel Manipulator	95
Table 4.38: Mean and Standard Deviation of Axial Tracking Errors for Diamond Motion of Parallel Manipulator.....	95
Table 4.39: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Diamond Motion of Parallel Manipulator.....	96
Table 4.40: Magnitude of Contour Tracking Performance for Diamond Motion of Parallel Manipulator.....	97
Table 4.41: Maximum Input Torques for Diamond Motion of Parallel Manipulator	98
Table 4.42: Mean and Standard Deviation of Axial Tracking Errors for Circular Motion of Parallel Manipulator.....	99

Table 4.43: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Circular Motion of Parallel Manipulator	100
Table 4.44: Magnitude of Contour Tracking Performance for Circular Motion of Parallel Manipulator.....	101
Table 4.45: Maximum Input Torques for Circular Motion of Parallel Manipulator	103
Table 4.46: Mean and Standard Deviation of Axial Tracking Errors for Horizontal Elliptical Motion of Parallel Manipulator	104
Table 4.47: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Horizontal Elliptical Motion of Parallel Manipulator.....	104
Table 4.48: Magnitude of Contour Tracking Performance for Horizontal Elliptical Motion of Parallel Manipulator.....	105
Table 4.49: Maximum Input Torques for Horizontal Elliptical Motion of Parallel Manipulator	107
Table 4.50: Mean and Standard Deviation of Axial Tracking Errors for Vertical Elliptical Motion of Parallel Manipulator	108
Table 4.51: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Vertical Elliptical Motion of Parallel Manipulator.....	108
Table 4.52: Magnitude of Contour Tracking Performance for Vertical Elliptical Motion of Parallel Manipulator.....	109
Table 4.53: Maximum Input Torques for Vertical Elliptical Motion of Parallel Manipulator...	111
Table 4.54: Magnitude of Contour Tracking Performance for High-Speed Motion of Serial Manipulator.....	112
Table 4.55: Magnitude of Contour Tracking Performance for High-Speed Motion of Parallel Manipulator.....	113
Table 4.56: Magnitude of Contour Tracking Performance for Motion with a Payload of Serial Manipulator.....	114
Table 4.57: Magnitude of Contour Tracking Performance for Motion with a Payload of Parallel Manipulator.....	114
Table 4.58: Magnitude of Contour Tracking Performance for Motion with High PD Gains for Serial Manipulator	116
Table 4.59: Magnitude of Contour Tracking Performance for Motion with High PD Gains for Parallel Manipulator.....	116

Table 4.60: Magnitude of Contour Tracking Performance for Motion with High SMC Gains for Serial Manipulator	117
Table 4.61: Magnitude of Contour Tracking Performance for Motion with High SMC Gains for Parallel Manipulator.....	118

LIST OF FIGURES

Figure 1.1: Estimated worldwide annual shipments of industrial robots.....	2
Figure 1.2: Estimated worldwide annual supply at year-end by industries	2
Figure 1.3: General Architecture of a Robotic Mechanical System	3
Figure 2.1: Block Diagram of Ideal PID Control	9
Figure 2.2: PD control with desired gravity compensation	11
Figure 2.3: Nonlinear Closed-Loop System	12
Figure 2.4: Illustration of NPD Control	13
Figure 2.5: Event-Driven PID Structure	14
Figure 2.6: Gain Scheduling Controller.....	17
Figure 2.7: MRAC Laws (a) Series High-Gain Scheme (b) Parallel Scheme.....	17
Figure 2.8: Self-Tuning Controller	18
Figure 2.9: Two types of Iterative Learning Control (a) Off-line ; (b) On-line.....	20
Figure 2.10: Interactions in a switching control system	22
Figure 2.11: Chattering about the sliding surface.....	23
Figure 2.12: Tracking and Contour Error	25
Figure 2.13: Error model for a linear contour.....	25
Figure 3.1: Block Diagram of ideal sliding mode control for a second order system.	32
Figure 3.2: Block diagram of an ideal second order PD-SMC system.....	34
Figure 4.1: Serial 3R Robotic Manipulator.....	44
Figure 4.2: Parallel 5R Robotic Manipulator.....	45
Figure 4.3: Zigzag Contour Task Space Positions and Velocities for Serial Manipulator	49
Figure 4.4: Zigzag Contour Joint Positions and Velocities for Serial Manipulator.....	50
Figure 4.5: Desired Zigzag End-effector Contour for Serial Manipulator	50
Figure 4.6: Diamond Contour Task Space Positions and Velocities for Serial Manipulator	51
Figure 4.7: Diamond Contour Joint Positions and Velocities for Serial Manipulator.....	51
Figure 4.8: Desired Diamond End-effector Contour for Serial Manipulator.....	52
Figure 4.9: Zigzag Contour Task Space Positions and Velocities for Parallel Manipulator	53
Figure 4.10: Zigzag Contour Joint Positions and Velocities for Parallel Manipulator.....	53
Figure 4.11: Desired Zigzag End-effector Contour for Parallel Manipulator	54

Figure 4.12: Diamond Contour Task Space Positions and Velocities for Parallel Manipulator ..	54
Figure 4.13: Diamond Contour Joint Positions and Velocities for Parallel Manipulator.....	55
Figure 4.14: Desired Diamond End-effector Contour for Parallel Manipulator.....	55
Figure 4.15: Circular Contour Task Space Positions and Velocities for Serial Manipulator	57
Figure 4.16: Circular Contour Joint Positions and Velocities for Serial Manipulator.....	58
Figure 4.17: Desired Circular End-effector Contour for Serial Manipulator	58
Figure 4.18: Elliptical Contour Task Space Positions and Velocities for Serial Manipulator	59
Figure 4.19: Elliptical Contour Joint Positions and Velocities for Serial Manipulator	60
Figure 4.20: Desired Elliptical End-effector Contour for Serial Manipulator	60
Figure 4.21: Circular Contour Task Space Positions and Velocities for Parallel Manipulator	61
Figure 4.22: Circular Contour Joint Positions and Velocities for Parallel Manipulator.....	62
Figure 4.23: Desired Circular End-effector Contour for Parallel Manipulator	62
Figure 4.24: Horizontal Elliptical Contour Task Space Positions and Velocities for Parallel Manipulator	63
Figure 4.25: Horizontal Elliptical Contour Joint Positions and Velocities for Parallel Manipulator	64
Figure 4.26: Vertical Elliptical Contour Task Space Positions and Velocities for Parallel Manipulator	64
Figure 4.27: Vertical Elliptical Contour Joint Positions and Velocities for Parallel Manipulator	65
Figure 4.28: Desired Parallel Manipulator End-effector Contours for (a) Horizontal Elliptical (b) Vertical Elliptical	65
Figure 4.29: Errors for Varying Proportional Gains for a Linear Contour.....	67
Figure 4.30: Errors for Varying Proportional Gains for a Nonlinear Contour	67
Figure 4.31: Errors for Varying Derivative Gains for a Linear Contour	68
Figure 4.32: Errors for Varying Derivative Gains for a Nonlinear Contour	68
Figure 4.33: Errors for Varying SMC Gains for a Linear Contour	69
Figure 4.34: Errors for Varying SMC Gains for a Nonlinear Contour.....	69
Figure 4.35: Errors for Varying Bandwidths for a Linear Contour	70
Figure 4.36: Errors for Varying Bandwidths for a Nonlinear Contour.....	70
Figure 4.37: Axial Tracking Error for Zigzag Motion of Serial Manipulator	73

Figure 4.38: End-Effector Task Space Tracking Error for Zigzag Motion of the Serial Manipulator	74
Figure 4.39: Contour Error for Zigzag Motion of the Serial Manipulator.....	75
Figure 4.40: Zigzag Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator	76
Figure 4.41: Input Torques for Zigzag Motion of Serial Manipulator	77
Figure 4.42: Axial Tracking Error for Diamond Motion of Serial Manipulator	78
Figure 4.43: End-Effector Task Space Tracking Error for Diamond Motion of the Serial Manipulator	79
Figure 4.44: Contour Error for Diamond Motion of the Serial Manipulator.....	80
Figure 4.45: Diamond Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator	80
Figure 4.46: Input Torques for Diamond Motion of Serial Manipulator.....	81
Figure 4.47: Axial Tracking Error for Circular Motion of Serial Manipulator	82
Figure 4.48: End-Effector Task Space Tracking Error for Circular Motion of the Serial Manipulator	83
Figure 4.49: Contour Error for Circular Motion of the Serial Manipulator.....	84
Figure 4.50: Circular Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator	85
Figure 4.51: Input Torques for Circular Motion of Serial Manipulator	86
Figure 4.52: Axial Tracking Error for Elliptical Motion of Serial Manipulator.....	87
Figure 4.53: End-Effector Task Space Tracking Error for Elliptical Motion of the Serial Manipulator	87
Figure 4.54: Contour Error for Elliptical Motion of the Serial Manipulator	89
Figure 4.55: Elliptical Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator	89
Figure 4.56: Input Torques for Elliptical Motion of Serial Manipulator	90
Figure 4.57: Axial Tracking Error for Zigzag Motion of Parallel Manipulator	92
Figure 4.58: End-Effector Task Space Tracking Error for Zigzag Motion of the Parallel Manipulator	93
Figure 4.59: Contour Error for Zigzag Motion of the Parallel Manipulator.....	94

Figure 4.60: Zigzag Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator	94
Figure 4.61: Input Torques for Zigzag Motion of Parallel Manipulator.....	95
Figure 4.62: Axial Tracking Error for Diamond Motion of Parallel Manipulator.....	96
Figure 4.63: End-Effector Task Space Tracking Error for Diamond Motion of the Parallel Manipulator	97
Figure 4.64: Contour Error for Diamond Motion of the Parallel Manipulator.....	98
Figure 4.65: Diamond Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator	98
Figure 4.66: Input Torques for Diamond Motion of Parallel Manipulator.....	99
Figure 4.67: Axial Tracking Error for Circular Motion of Parallel Manipulator	100
Figure 4.68: End-Effector Task Space Tracking Error for Circular Motion of the Parallel Manipulator	101
Figure 4.69: Contour Error for Circular Motion of the Parallel Manipulator.....	102
Figure 4.70: Circular Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator	102
Figure 4.71: Input Torques for Circular Motion of Parallel Manipulator.....	103
Figure 4.72: Axial Tracking Error for Horizontal Elliptical Motion of Parallel Manipulator....	104
Figure 4.73: End-Effector Task Space Tracking Error for Horizontal Elliptical Motion of the Parallel Manipulator	105
Figure 4.74: Contour Error for Horizontal Elliptical Motion of the Parallel Manipulator	106
Figure 4.75: Horizontal Elliptical Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator	106
Figure 4.76: Input Torques for Horizontal Elliptical Motion of Parallel Manipulator.....	107
Figure 4.77: Axial Tracking Error for Vertical Elliptical Motion of Parallel Manipulator	108
Figure 4.78: End-Effector Task Space Tracking Error for Vertical Elliptical Motion of the Parallel Manipulator	109
Figure 4.79: Contour Error for Vertical Elliptical Motion of the Parallel Manipulator	110
Figure 4.80: Vertical Elliptical Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator	110
Figure 4.81: Input Torques for Vertical Elliptical Motion of Parallel Manipulator	111

Figure 4.82: Contour Error for High-Speed Motion of the Serial Manipulator.....	112
Figure 4.83: Contour Error for High-Speed Motion of the Parallel Manipulator.....	113
Figure 4.84: Contour Error for Motion with a Payload of the Serial Manipulator	114
Figure 4.85: Contour Error for Motion with a Payload of the Parallel Manipulator	115
Figure 4.86: Contour Error for Motion with a High PD Gains for Serial Manipulator	116
Figure 4.87: Contour Error for Motion with a High PD Gains for Parallel Manipulator	117
Figure 4.88: Contour Error for Motion with a High SMC Gains for Serial Manipulator.....	118
Figure 4.89: Contour Error for Motion with a High SMC Gains for Parallel Manipulator.....	118
Figure 4.90: Serial Manipulator Mean Contour Error Magnitude Performance Comparison	121
Figure 4.91: Parallel Manipulator Mean Contour Error Magnitude Performance Comparison .	121

ACRONYMS

DES	<i>Discrete Event System</i>
EDC	<i>Event-Driven Control</i>
ILC	<i>Iterative Learning Control</i>
MRAC	<i>Model Reference Adaptive Control</i>
NC	<i>Numerical Control</i>
NPD	<i>Nonlinear Proportional-Derivative</i>
NPD-LC	<i>Nonlinear Proportional-Derivative Learning Control</i>
PD	<i>Proportional-Derivative</i>
PD-OLC	<i>Proportional-Derivative-Type On-line Learning Control</i>
PD-SMC	<i>Proportional-Derivative Sliding Mode Control</i>
PI	<i>Proportional-Integral</i>
PID	<i>Proportional-Integral-Derivative</i>
RC	<i>Repetitive Control</i>
SCARA	<i>Selective Compliance Articulated Robot Arm</i>
SMC	<i>Sliding Mode Control</i>
STC	<i>Self-Tuning Control</i>

NOMENCLATURE

B	<i>Matrix of motor viscous damping coefficients</i>
$C(\dot{q}, q)$	<i>Matrix of centrifugal-Coriolis terms</i>
$\hat{C}(\dot{q}, q)$	<i>Estimated matrix of centrifugal-Coriolis terms</i>
$D(t, \dot{q}, q)$	<i>Vector of friction, uncertainty, disturbance terms</i>
$G(q)$	<i>Vector of gravitational torques/forces</i>
$\hat{G}(q)$	<i>Estimated vector of gravitational torques/forces</i>
J	<i>Matrix of motor inertias</i>
K_D	<i>Derivative gain matrix</i>
K_I	<i>Integral gain matrix</i>
K_P	<i>Proportional gain matrix</i>
K_S	<i>SMC gain matrix</i>
L_i	<i>Length of i^{th} link</i>
$M(q)$	<i>Inertia matrix</i>
$\hat{M}(q)$	<i>Estimated inertia matrix</i>
$V(e(t), \dot{e}(t))$	<i>Lyapunov function</i>
e	<i>Position tracking error</i>
\dot{e}	<i>Velocity error</i>
\ddot{e}	<i>Acceleration error</i>
g	<i>Gravitational force</i>
x_{ee}	<i>End-effector x-coordinate</i>
y_{ee}	<i>End-effector y-coordinate</i>
q	<i>Joint/axis (angular) position or end-effector pose</i>
\dot{q}	<i>Joint/axis (angular) velocity or end-effector pose velocity</i>
\ddot{q}	<i>Joint/axis (angular) acceleration or end-effector pose velocity</i>
s	<i>Sliding surface</i>
$s_c(t)$	<i>Trajectory definition parameter</i>
t	<i>Time</i>
ε	<i>Contour error</i>
λ	<i>SMC bandwidth</i>
$\lambda_M(M)$	<i>Largest eigenvalue of M</i>
$\lambda_m(M)$	<i>Smallest eigenvalue of M</i>
τ	<i>Joint torque/force</i>
ϕ	<i>Boundary layer</i>

Chapter 1 : INTRODUCTION

1.1 BACKGROUND

Development of early industrial robots occurred in the 1960s as a result of the convergence of two technologies: numerical control (NC) machines for precise manufacturing, and tele-operated machines for the remote handling of hazardous material. Compared to its predecessors, the first robot manipulators were characterized by their versatility, adaptability, positioning accuracy, and execution repeatability (Siciliano, Sciavicco, Villani, & Oriolo, 2009). In the following decades, industrial robots gained widespread popularity as essential components for the realization of automated manufacturing systems. The increasingly wide range of applications in the manufacturing industry to where robotics technology is being applied is the result of several factors, including the reduction of manufacturing costs, the increase in productivity, improvement of production quality, and the reduction or the elimination of harmful tasks for the human operator.

In recent years, following an interruption caused by the global financial crisis in 2009, the trend towards automation continued. In fact, 2011 was the most successful year for industrial robots since 1961 (International Federation of Robotics, 2013). In that year, industrial robot sales increased by 38% to 166,028 units. This was by far the highest level ever recorded for a single year. According to the IFR Executive Summary, the predominant user of industrial robots is the automotive/motor vehicle industry.

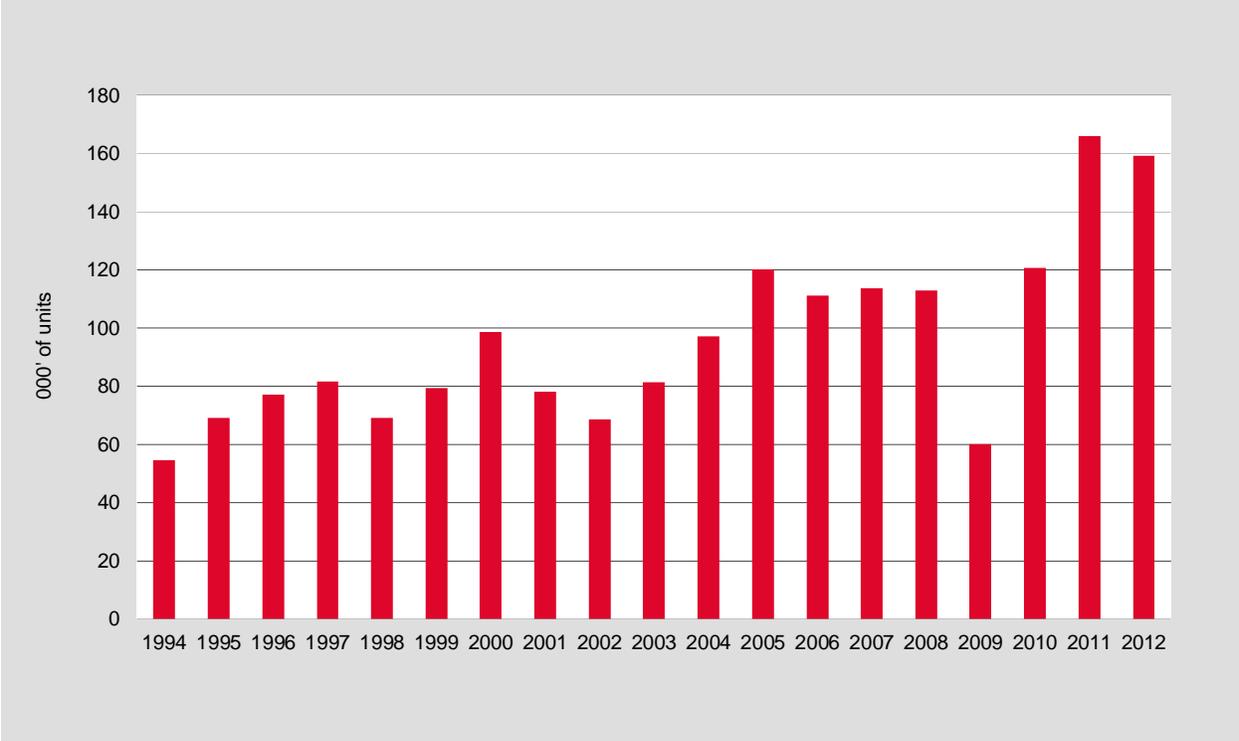


Figure 1.1: Estimated worldwide annual shipments of industrial robots (International Federation of Robotics, 2013).

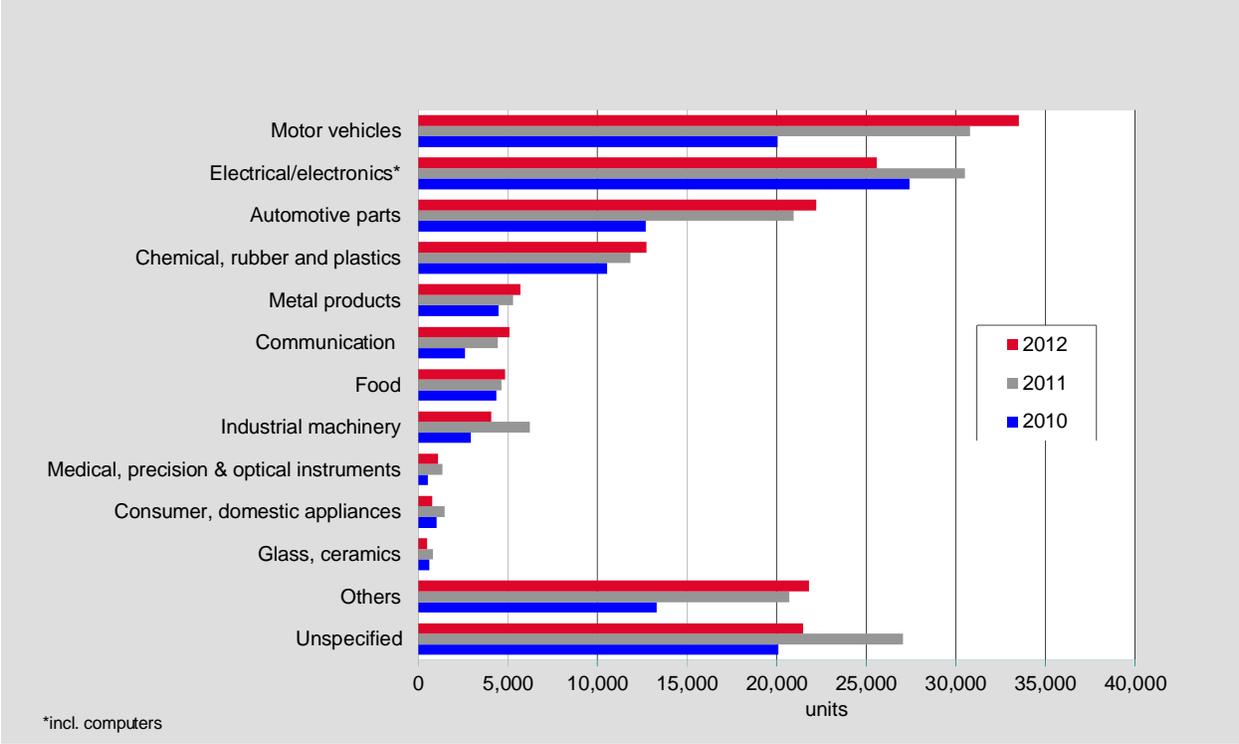
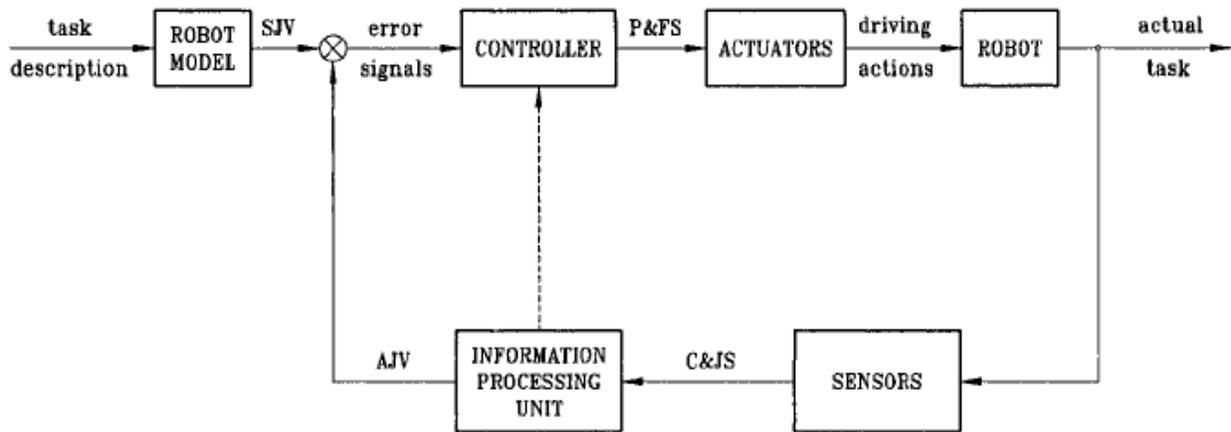


Figure 1.2: Estimated worldwide annual supply at year-end by industries (International Federation of Robotics, 2013).

In light of the continued trend towards automation over a wide range of applications, industrial robots display significant characteristics of versatility and accuracy. By virtue of their programmability, industrial robots remain essential components to many automated systems, see Figure 1.2.

Industrial robots are a subset of the broader category of robotic systems. A robotic system is composed of several subsystems, including a mechanical subsystem of rigid or deformable bodies, a sensing subsystem, an actuation subsystem, a controller, and an information-processing subsystem, see Figure 1.3. These subsystems communicate among themselves via interfaces, whose function is decoding and transmitting data between mediums. Robots with a fixed base are classified as robot manipulators (Siciliano, Sciavicco, Villani, & Oriolo, 2009).



- SJV: synthesized joint variables (angles and torques)
- P&FS: position and force signals
- C&JS: Cartesian and joint signals
- AJV: actual joint variables (angles and torques)

Figure 1.3: General Architecture of a Robotic Mechanical System (Angeles, 2007).

The manipulator's sequence of rigid bodies (*links*) is interconnected by means of articulation (*joints*). The fundamental structure of a manipulator is a serial, or open kinematic

chain. Alternatively, a manipulator can contain closed kinematic chains when a sequence of links forms a loop. Robotic manipulators can have several different geometries with varying levels of dexterity and stiffness. Serial manipulators have good dexterity and have straightforward dynamics while closed chain manipulators, of particular interest those with parallel geometries, have the fundamental advantage of high structural stiffness with respect to serial manipulators at the cost of reduced workspaces. The industrial application of the robotic manipulator is a factor that determines its geometry.

In all robot applications, the completion of a generic task requires the execution of a specified prescribed motion. This execution is entrusted to the control system. Due to the significance of programmable robots in industry, the research and development of effective control laws is important. Each control law is designed to compensate for the inherent imprecisions, disturbances, and other errors presented in the system. The causes for these imprecisions include unmodeled friction and vibrational dynamics, hardware deficiencies, process-generated disturbances, and other inaccuracies of the system dynamics (Slotine & Li, 1988).

For many industrial applications, good trajectory tracking control is very important. Depending on the specific application or environment, the uncertainty or parameter fluctuation may call for increased robustness of the controller. It is important to take each of these factors as well as the previously mentioned imprecisions into account when designing or implementing a control law.

The scope of this thesis involves the proposal of a hybrid controller that employs PD control as well as the nonlinear switching component of sliding mode control. It will demonstrate

that this proposed PD-SMC law is a powerful and robust control law for improving the trajectory and contour tracking performance of robot manipulators. This is a crucial area of research since the increasing trend in automation calls for control laws that provide more accurate and robust performance.

1.2 MOTIVATION AND OBJECTIVES

The motivation behind this research is to improve the tracking performance of robotic manipulators and, through the reduction of tracking error, seek to improve contour tracking performance as well. This thesis will propose a new hybrid control law based on PD and sliding mode control theory to provide a powerful and robust control law. By providing accurate control on the joint level, the overall tracking performance of the manipulator will be improved. The overall goals of the proposed hybrid PD-SMC law are:

1. To provide an alternative to standard PID/PD control laws.
2. To incorporate the advantages of nonlinear control into an easily implemented control law.
3. To improve performance by reducing joint tracking errors in comparison to traditional PD control.
4. To provide robust tracking performance for systems with imprecisely modeled dynamics or fluctuations in parameters.
5. To provide a model-free alternative to sliding mode control laws that require knowledge of the dynamics of the system.

In this thesis, the proposed hybrid PD-SMC law will be verified through simulations for linear and nonlinear contours. To demonstrate the performance and improvements, standard PD

and SMC laws will also be simulated and the tracking and contour errors will be compared for various motions and loading cases. To fulfill these goals, the following objectives will be achieved:

1. Model the dynamics of a simplified RRR serial planar manipulator and a five-bar planar parallel manipulator.
2. Formulate a hybrid PD sliding mode control law.
3. Verify the stability of the proposed controller using the Lyapunov method.
4. Perform a comparison of the proposed hybrid PD-SMC law and the conventional PD and SMC laws at the joint and end-effector level.
5. Extend the comparison to high-speed and payload cases in order to verify the robustness of the proposed controller.

1.3 ORGANIZATION OF CONTENTS

The remainder of the thesis is organized as follows:

Chapter 2 is a literature review of previous research on the control of robotic manipulators. It will summarize the various methods employed in improving performance through the reduction of tracking error and contour error. It will also give a brief inspection on some advanced control techniques and the methodology behind the hybridization of control laws.

Chapter 3 presents the mathematical formulation of the dynamics of the manipulators that will be simulated. From these dynamics models, the standard SMC law will be formulated for each manipulator. This chapter will also contain the realization of the proposed hybrid PD-SMC law

and the fulfillment of stability analysis of the proposed PD-SMC controller using the Lyapunov method.

Chapter 4 contains the simulation results comparing the performance of the PD, SMC and the proposed PD-SMC law for various linear and nonlinear trajectories. It also provides a guideline for the selection of the control parameters of the PD-SMC law.

Chapter 5 provides a summary of results, conclusions, and proposals for future academic endeavors with the hybrid control law.

Chapter 2 : LITERATURE REVIEW

If the dynamic model of a controlled system is known perfectly, an idealized controller may be implemented in order to achieve very good system performance. Practically, however, system parameters may be difficult to determine and are never known perfectly. In addition, the external payloads and disturbances introduced to the system can make it very difficult to implement an ideal controller. One of the reasons why adaptive and robust controllers are developed is to overcome these dynamic model uncertainties and external disturbances (Sage, De Mathelin, & Ostertag, 1999).

On the other hand, there exists some control methods that do not require previous knowledge of system dynamics, which makes them easily implemented. However, they often do not account for nonlinearities in the dynamics of the system. In applications that require more precise tracking performance, these control laws may not be sufficient. Therefore, different types of control systems were developed for different applications. In this chapter, some popular and commonly applied control systems are reviewed.

2.1 PD/PID CONTROL

Because of its simple structure, the clear physical meaning of each control gain, and easy implementation, PID control (as well as PI and PD control) has garnered widespread popularity amongst engineers, with more than 90% of all control loops using some form of PID (Åström & Hägglund, 2001). PD/PID has been very widely employed in many industrial applications, such as robotic control, process control, and automatic systems.

2.1.1 General PID Control

A typical PID controller has the following form:

$$T(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \dot{e}(t) \quad (2.1)$$

where K_P , K_I , and K_D are the proportional, integral, and derivative gain matrices of the control system, respectively, and $T(t)$ is the torque.

The tracking error, which can be a vector, is defined as:

$$e(t) = q_d - q \quad (2.2)$$

Although the basic structure of the PID controller is quite simple, as shown in Figure 2.1, it has been shown that with appropriately selected control gains, a simple linear and decoupled PD controller may have acceptable tracking performances for many applications (Qu, 1994).

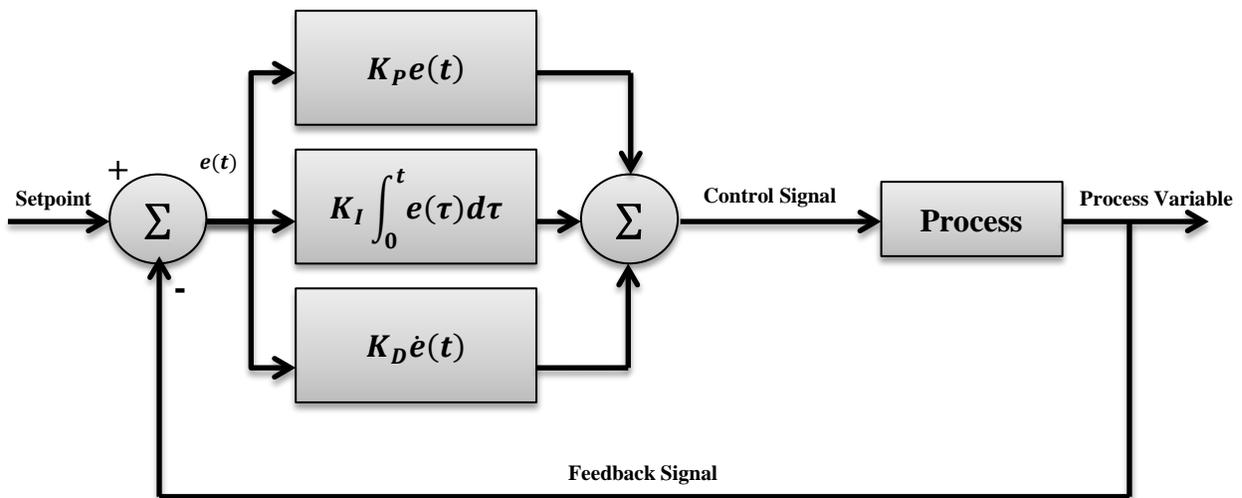


Figure 2.1: Block Diagram of Ideal PID Control

PID control has been applied for the tracking control of robotic manipulators (Hsu, Chiu, & Tsai, 2011; Pervozvanski & Freidovich, 1999) and although PID is effective when

implemented as a position controller, care must be taken into the tuning of the control gains through the use of a theoretical backing.

A theorem proposed by Kawamura *et al.* proves that a robot equipped with a linear PD feedback loop at each joint can track a desired trajectory described by time functions if the velocity feedback gain is set sufficiently large (Kawamura, Miyazaki, & Arimoto, 1988). There have been several other research endeavors for PD control on its global stability for trajectory tracking with robotic manipulators (Qu, 1994; Chen, Chen, Wang, & Woo, 2000). The high control gains of PD control, however, may cause vibration problems due to overreaction and amplification of the measurement noise. Also, there are still certain cases for which standard PID control is inappropriate due to its inability to adjust to system dynamics.

2.1.2 PD with Desired Gravity Compensation

Although the use of PID control is extensive for the control of robotic manipulators, the choice of PID gains relies on relatively complex formulae in order to ensure global asymptotic stability (Arimoto & Miyazaki, 1984; Qu & Dorsey, 1991). The landmark work of Takegaki and Arimoto (1981) provided a robust position control that is easily tuned for global asymptotic stability. The PD control law with desired gravity compensation for setpoint control proposed by Takegaki and Arimoto (1981) can be written as:

$$\tau = K_p e - K_D \dot{q} + g(q_d) \quad (2.3)$$

where τ represents the torque in this formula and $g(q_d)$ is the vector of gravitational torques of the system at the desired joint angles. Graphically, this control law is represented in Figure 2.2 and $K_v = K_D$.

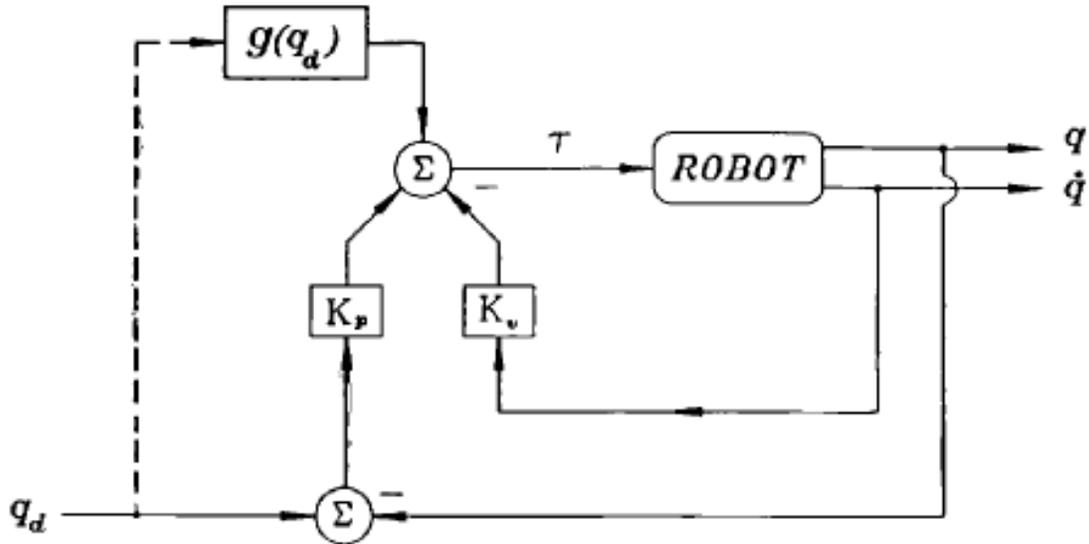


Figure 2.2: PD control with desired gravity compensation (Kelly, 1997).

The PD with desired gravity compensation control law is able to drive the manipulator in such a way that the robot joints are asymptotically placed at their desired values. It does this independently of the initial position and velocity. Although this controller has a simple structure, it still requires knowledge of the system dynamics in order to calculate the gravitational torque vector terms. This desired gravity component can be seen as a feedforward term that helps the control law compensate for the system dynamics.

2.1.3 Nonlinear PD/PID Control

One of the weaknesses of standard PD/PID control is that it may suffer in performance due to the linearity of the control law and its poor compensation for the inherent nonlinearity of many practical dynamic systems (and sensors). One method of refining the PID control law was proposed by Rugh (1987), where he designed a law by the extended linearization technique in which the assumed nonlinear plant and sensor, as shown in Figure 2.3, are linearized about every closed-loop set point with the appropriate PID controller. In his proposed design, the gains of the controller in the form of PID are appropriately defined functions of the error state.

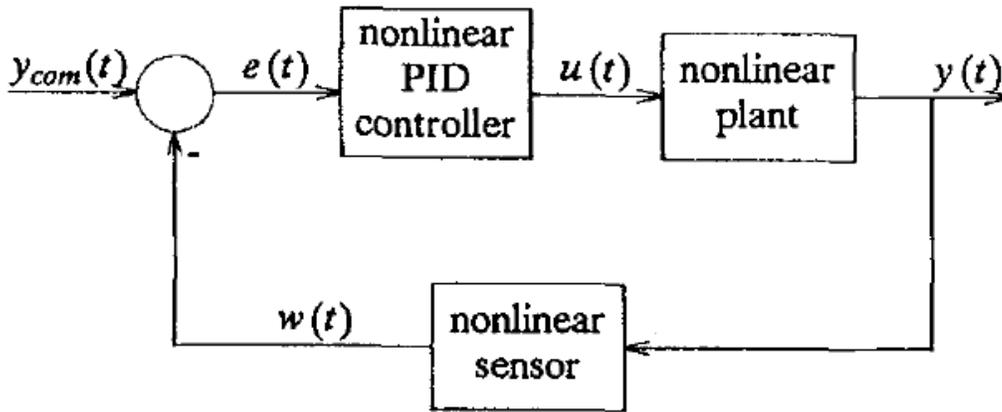


Figure 2.3: Nonlinear Closed-Loop System (Rugh, 1987).

This Nonlinear PD (NPD) control is, in general, a control where the gains are functions of tracking errors. Consider the general system equation:

$$\ddot{x} + c(\dot{x}, x) = u \quad (2.4)$$

where \dot{x} and x are system states and $c(\dot{x}, x)$ is a nonlinear, state-dependent term. According to Xu *et al.*, for force control of a setpoint task, we have the NPD control in the form:

$$u = Ke_f + B\dot{e}_f + f_d + \hat{c}(\dot{x}, x) \quad (2.5)$$

where K and B are gains, f_d is the desired force, $e_f = f_d - f$, and $\hat{c}(\dot{x}, x)$ is the estimate of $c(\dot{x}, x)$. The NPD's control action depends on whether the system is moving towards or away from the desired setpoint as follows (Xu, Hollerbach, & Ma, 1995):

- Moving away from setpoint: increase gains to stop the system
- Moving toward setpoint: decrease gains to minimize the residual energy at the goal

This can be illustrated graphically in Figure 2.4. Although the study presented by Xu *et al.*, was for force control, their methods are equally applicable to position control. Several other previous studies were conducted involving NPD control, however many were applied only to

point-set control of linear systems (Armstrong & Wade, 2000; Seraji, 1998). Ouyang (2005) extended NPD to a form of adaptive NPD learning control designed for repetitive tasks of robotic mechanisms.

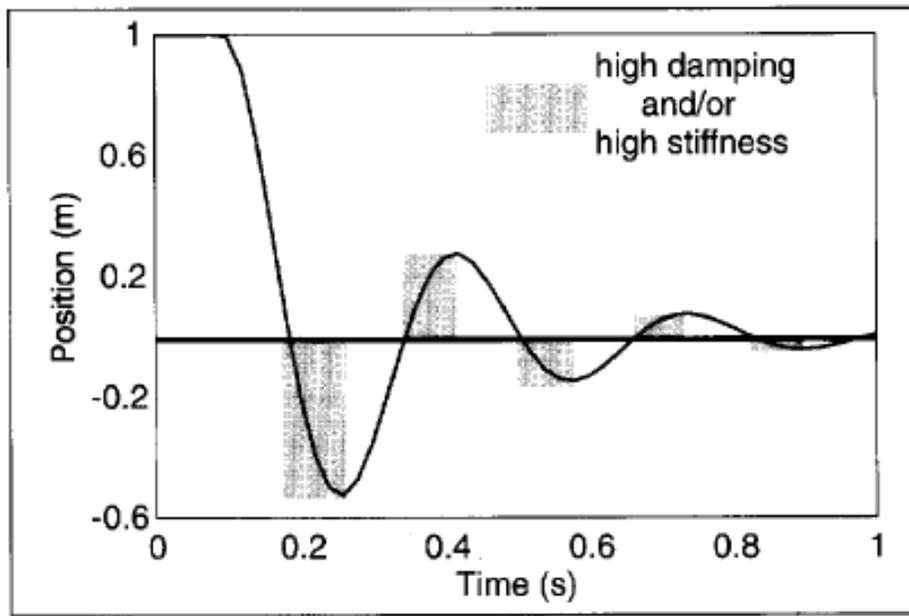


Figure 2.4: Illustration of NPD Control (Xu, Hollerbach, & Ma, 1995).

The NPD control laws provide a method of compensation for the nonlinear system dynamics of many mechanical systems, but may demand some involved stability requirements. Also, as with other forms of control with feedforward terms, knowledge of the system dynamics is required for this approach.

2.2 OTHER CONTROL METHODS

The previous section introduced some of fundamental control concepts used for robotic manipulators over the course of study in industry. This section furthers the discussion to some more advanced methods of control, including event-driven control (EDC), adaptive control, and learning control.

2.2.1 *Event-Driven Control*

The majority of research in control theory and engineering considers time-triggered or periodic control systems where signals are acquired at a fixed sample rate in an attempt to represent a continuous time signal (Heemels, Sandee, & Van Den Bosch, 2008). There are cases, however, where it is of interest to consider event-driven control systems where instead of time-triggered sampling, samples are acquired at the occurrence of an event. The traditional time-triggered control methods have the controller performing a new control action at the prescribed time interval even when there is nothing happening in the system. This can lead to the unnecessary utilization of system resources (Årzén, 1999).

Årzén (1999) proposed a simple event-based PID controller that consists of two parts: (1) an event detection part that uses time-triggered sampling on a sampling interval that is the same as the sampling interval of the corresponding time-triggered PID controller, and (2) an event-triggered PID controller that uses the output of the event detection part. This output sends a request to the PID control algorithm that a new control signal should be calculated. This is represented graphically in Figure 2.5.

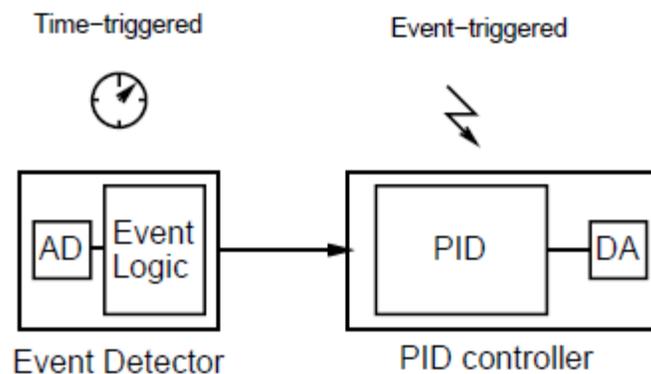


Figure 2.5: Event-Driven PID Structure (Årzén, 1999).

Typically, an embedded controller on a real-time operating system uses available central processing unit (CPU) time that is shared between independently running tasks. Using CPU resources when there is nothing of significance happening in the process is clearly a waste of resources. Experiments conducted by Årzén with this control structure indicate significant reduction in CPU utilization with only a minor control performance penalty. In work conducted by Wang *et al.* (2011), event-driven controllers are shown to have practical application to motion control, including vehicle longitudinal control and quadrotor control. In their study, the use of event-triggered controllers yielded strong robustness with the need for only few computing resources. Additionally, small system vibrations in the quadrotor were eliminated while the number of actuation steps was diminished.

2.2.2 *Adaptive Control*

From looking at the previous section, we can generally see that for a given form of controller (i.e. PID), there is a trend for better performance with a more intimate knowledge of the plant dynamics. In fact, there are many current techniques that rely on having a good understanding of the plant under study in some form or another (Sastry & Bodson, 1989). However, there are a number of instances where either the system to be controlled is far too complex, or the basic physical processes of the system are not fully understood. Adaptive control compensates for this by obtaining a progressively better understanding of the plant dynamics to be controlled. Thus, adaptive control can be viewed as being composed of two parts (Craig, 1988):

1. an identification portion that identifies the parameters that appear in the controller of the plant, or the parameters of the plant itself and;
2. a control law portion, which implements the control law that is in some way a function of the identified parameters.

If the identification of the plant is periodically updated on the basis of previous estimates and new data, then the system identification is said to be recursive and the identification and control may be performed concurrently to one another. From the above discussion, it can be said that adaptive control is designed to compensate for the time-varying parameters of the controlled system. It should also be noted that system identification can also be targeted at the determination of whether the plant is linear or nonlinear, finite or infinite dimensional, and has continuous or discrete event dynamics (Sastry & Bodson, 1989). If these have already been determined, then the adaptive control law will be limited to the parametric system identification or *parametric adaptive control*, as was discussed.

In common frameworks there are three schemes for parametric adaptive control: (1) gain scheduling, (2) model reference adaptive control (MRAC), and (3) self-tuning regulators (STC) (Åström K. J., 1983). Gain scheduling was one of the earliest and most intuitive approaches to adaptive control. With gain scheduling, auxiliary process variables, other than those from the plant used as feedback, were found that correlate well with process dynamics. The advantage of this type of control is that the parameters can be changed as quickly as the auxiliary measurement. Although gain scheduling was popular in practice, it does have the drawback of being an open-loop adaptation scheme with no real intelligence. Also, the extent of design that is required may be enormous, depending on the system. Gain scheduling is represented graphically in Figure 2.6.

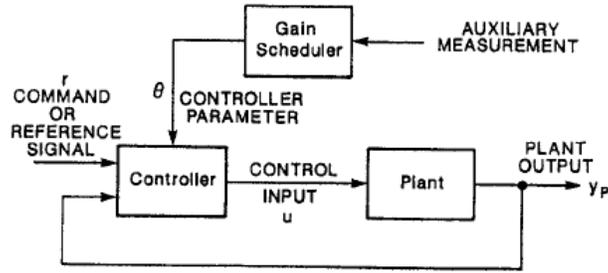


Figure 2.6: Gain Scheduling Controller (Sastry & Bodson, 1989).

There are two types of MRAC laws: series high gain scheme and parallel scheme, both represented schematically in Figure 2.7. For each of these schemes, the desired performance of the closed-loop system is specified through a reference model and the adaptive system attempts to make the plant output match the reference model. With the parallel scheme, however, the inner or regulator loop acts as an ordinary control loop consisting of the plant and regulator, and the outer adaptation loop adjusts the parameters of the regulator in such a way to drive the error between the model and plant output to zero.

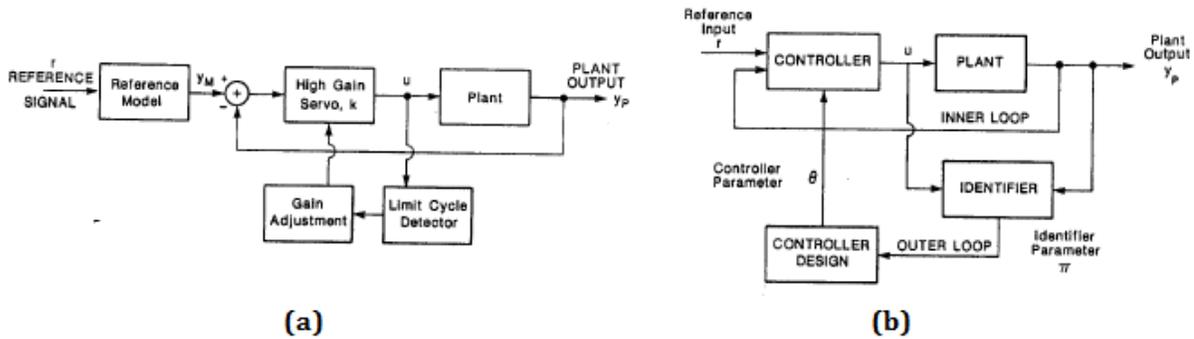


Figure 2.7: MRAC Laws (a) Series High-Gain Scheme (b) Parallel Scheme (Sastry & Bodson, 1989).

Self-tuning regulators, as shown in Figure 2.8, have an identifier or estimator that produces a set of estimated parameters based on past plant inputs and outputs. The controller then uses these to produce the current control input for the plant.

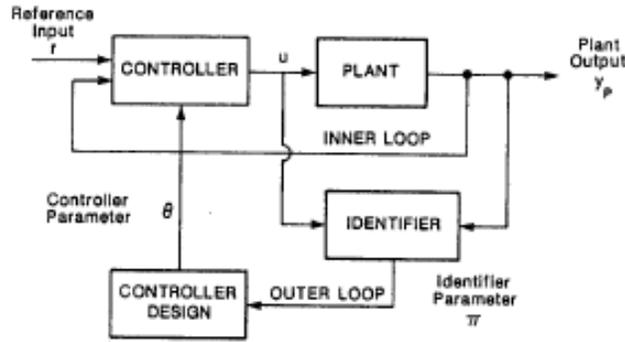


Figure 2.8: Self-Tuning Controller (Sastry & Bodson, 1989).

With regard to improving trajectory tracking performances of robotic manipulators, there has been significant research on the basis of adaptive control, including the control for direct-drive SCARA robots (Dessaint, Saad, Hébert, & Al-Haddad, 1992). Furthermore, Tomei (1991) and Kuc & Han (2000) have employed adaptive control laws to PD and PID loops, respectively. This implementation of adaptive control laws with existing laws owes to the flexibility of adaptive controls. Slotine (1991) gives further details of the advantages and drawbacks of adaptive controls.

2.2.3 *Learning Control*

Learning control is similar to adaptive control where it uses gathered information from the past processes in order to determine its control performance for the future. The difference is that adaptive control depends on very recent history and reacts only to the current state, while a learning controller depends on long term history and remembers previous states with their appropriate responses (White & Sofge, 1992).

Learning control operates in the context of an objective function and receives performance feedback. This feedback is used to determine how appropriate the current behaviour is in the context of the objective function. It is not necessary to have a parametric model for the system with learning control. In addition, the uncertainty can be compensated since the system

“learns” as it is propagated through trials. Both adaptive control and learning control are used to simplify the process of implementing the controller and to improve system performance by improving reliability of the system.

2.3 HYBRIDIZATION OF CONTROL LAWS

The previous section established some details for a few methods of advanced control. For several of them, it involved PID or other types of controllers in its implementation. This can be seen as the hybridization of several control laws in order to achieve a more desirable performance than that of which the individual control law is capable. This section will review in more detail the concept of control hybridization on iterative learning control (ILC), which can be used as a hybrid control system. Some specific hybrid control systems are reviewed including switching control, and the sliding mode control associated with the thesis.

2.3.1 *Iterative Learning Control*

Iterative learning control is based on the idea that the performance of a system can be improved by learning previous executions of the same task over and over again. Through these repeated actions, the properties of unknown dynamics may be revealed during the completion of a task. Although ILC emulates human learning, ILC differs from other learning-type control algorithms, such as adaptive control. Adaptive control modifies the controller, a system, where ILC modifies the control input, a signal (Bristow, Tharayil, & Alleyne, 2006).

Arimoto *et al.* (1984) defined ILC as class of control algorithms to achieve asymptotic zero tracking error by an iterative process. Therefore, the objective of ILC is to incorporate error information into the control for subsequent iterations, thereby improving performance. In doing so, low transient tracking error and high performance are achieved in spite of model uncertainties

and disturbances. A typical ILC in the time domain is under a simple closed-loop or open-loop. These two types of ILC can be classified as either off-line or on-line, respectively, represented schematically in Figure 2.9. ILC can be further classified into P-Type, D-Type, and PID Type (Ouyang, 2005). Several research papers have proposed controllers of these types including an adaptive PID learning control for the control of robotic manipulators where the control gains are functions of not only previous iterations, but are also complex functions of the errors of the current iteration (Kuc & Han, 2000). This is an example of creating a hybrid control algorithm by taking explicitly the characteristics of several different algorithms.

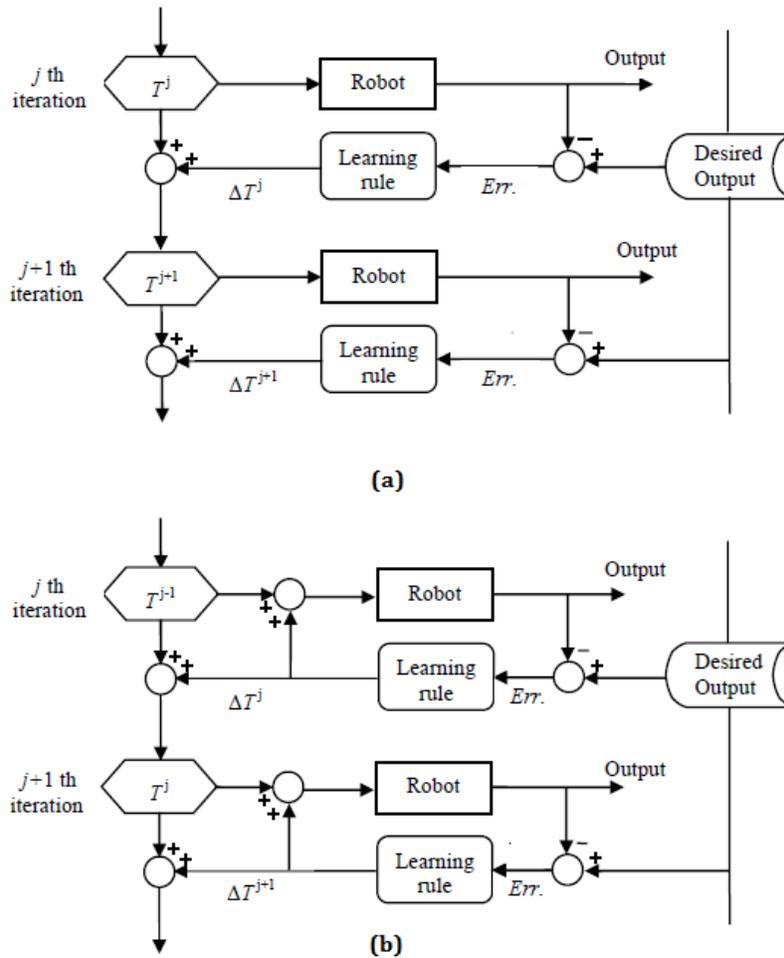


Figure 2.9: Two types of Iterative Learning Control (a) Off-line ; (b) On-line (Ouyang, 2005).

Ouyang (2005) proposed a hybrid PD-type online learning control (PD-OLC) for the tracking of a class of nonlinear time-varying systems. This control was demonstrated to be robust with respect to uncertainties and disturbances on the controlled system. He also formulated a hybrid adaptive nonlinear PD learning control (NPD-LC) which incorporated both feedback plus feedforward control, while classical ILC is only feedforward control. Because adaptive NPD-LC is an on-line learning control it is expected to have faster convergent speed than classical ILC.

2.3.2 *Switching Control*

Switching control systems are those that have discrete event dynamics as well as continuous time dynamics. It should be noted that in the field of controls research, switching control systems are also called hybrid control systems. The distinction should be made that the thesis uses the term hybrid control systems to denote control systems in general that include two or more contributing control laws.

Switching control systems are characterized by the interaction of continuous parts, governed by differential equations, and discrete parts that are described by finite states machines, if-then-else rules, and propositional and temporal logic (Ouyang, 2005). The interface between the components is shown in Figure 2.10, where DES stands for discrete event system, r is the input, z is the output and h_i means the “hybrid” part of the system. The index i is analogous to a time index for the discrete system in that it specifies the order of states or events in the system (Stiver & Antsaklis, 1992).

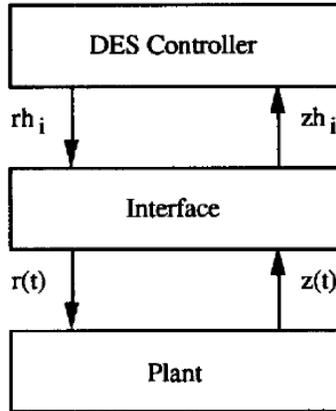


Figure 2.10: Interactions in a switching control system (Stiver & Antsaklis, 1992).

These types of controllers produce switches between a countable, possibly finite, number of fixed controllers until stability is detected. As such, they are theoretically attractive for providing performance with asymptotic stability for a wide array of various plants. They do, however, require a long time to search for stabilizing feedback and as such, provide poor transient properties (Antsaklis & Nerode, 1998).

2.3.3 *Sliding Mode Control*

Sliding mode control has been studied to extent in order to handle the problems of nonlinear dynamic control systems with modeling uncertainties, time-varying parameter fluctuations, and external disturbances (Slotine & Li, 1991). For this control law, a sliding mode surface is defined for an n^{th} -order dynamic system as:

$$s = \left(\frac{d}{dt} + \lambda \right)^{n-1} \quad (2.6)$$

where λ is strictly a positive constant. With this definition, following the process outlined by Slotine & Li (1991), it can be seen that an n^{th} -order tracking problem can be replaced by a first-order stabilization problem of keeping s at zero. By choosing an appropriate control law, the system trajectories can be constrained to point to the time-varying surface $S(t)$. In other words, it

makes the surface an invariant set. Once on the surface the system is said to be in sliding mode. If the dynamics of the system are exactly known, then by solving the dynamics while in sliding mode ($\dot{s} = 0$), an expression for the control force u , called the equivalent control u_{eq} , can be obtained and is interpreted as the continuous control law that would maintain perfect tracking $\dot{s} = 0$.

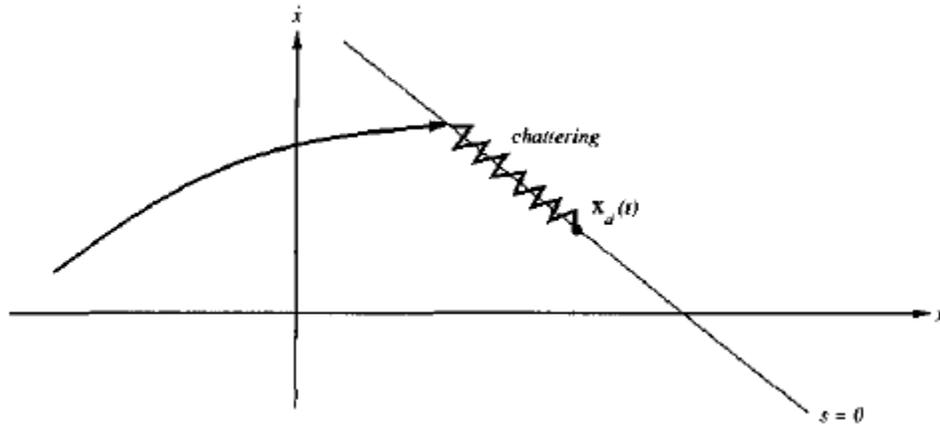


Figure 2.11: Chattering about the sliding surface (Slotine & Li, Applied Nonlinear Control, 1991).

Because of inherent uncertainty in the system model, only the best estimate for equivalent control is obtained, denoted \hat{u} . Perhaps one of the most intriguing aspects of SMC is the discontinuous nature that arises from this. In order to satisfy the sliding condition, a term that is discontinuous across the surface $s = 0$ is added to obtain the control law:

$$u = \hat{u} - k \operatorname{sgn}(s) \quad (2.7)$$

where k is a control parameter and $\operatorname{sgn}()$ is the sign function. It can be interpreted that since the control system deliberately switches between distinctively different structures based on the state of the system and some predefined rules, that SMC is a kind of switching or hybrid control (Ouyang, 2005).

Ideally, the switching of the controller happens instantaneously, resulting in a system trajectory that stays in sliding mode, as desired. However, in reality the finite sampling rate will cause the trajectory to oscillate about the sliding surface, as shown in Figure 2.11. This so-called chatter is in general undesirable since it can excite high-frequency dynamics of the system for which the control is unaccounted. The introduction of a boundary layer can smooth the trajectory (Slotine & Li, 1991). In addition to general SMC, there has been research on advanced control strategies involving SMC including using genetic algorithms to optimize control gains (Li *et al.*, 1996) and fuzzy chatter-free control (Allamehzadeh & Cheung, 2002).

2.4 TRACKING AND CONTOUR ERROR

2.4.1 *Tracking and Contour Error*

The performance of industrial manipulators is often related to two parameters: tracking error and contour error. Tracking error is the result of the individual axial tracking performance of each actuator. Another way to describe this is how well the controller for each axis can track the desired reference position or other state variable.

Mathematically, the tracking error can be expressed as

$$e = R - P \quad (2.8)$$

where R is the reference, or desired, position, and P is the actual position.

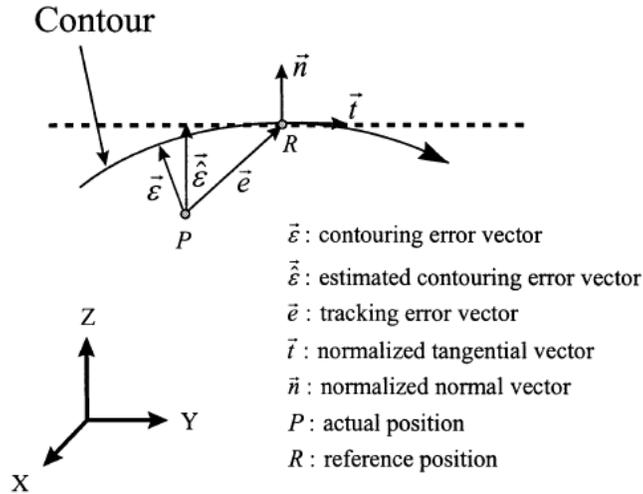


Figure 2.12: Tracking and Contour Error (Yeh & Hsu, 2003).

Contour error, however, can be defined as the component of error that is orthogonal to the desired path or contour. In conventional multi-axis contouring systems, contour errors arise from the differences in loop parameters, disturbance loads, and the complexity of the contour shape. Additional error sources due to mechanical hardware can include backlash and friction (Koren & Lo, 1991).

2.4.2 Contour Error Calculation

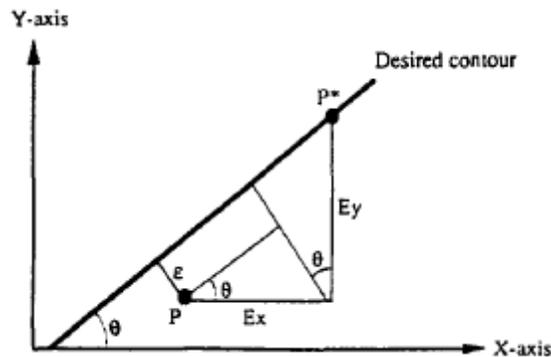


Figure 2.13: Error model for a linear contour (Koren & Lo, 1991).

For many simple contour shapes, such as lines and circles, contour error is easy to determine. For linear contours, the contour error ϵ is given by:

$$\varepsilon = -e_x \sin \theta + e_y \cos \theta \quad (2.9)$$

where e_x and e_y are the tracking error at the actual position P , and θ is the angle formed by the contour and the x-axis at the desired position P^* . Note that Figure 2.13 uses capitals to denote tracking error.

For circular contours, the Cartesian equation of a circle can be used to find the contour error as follows

$$\varepsilon = \sqrt{(P_x - x_0)^2 + (P_y - y_0)^2} - R \quad (2.10)$$

where R is the radius of the circle, (x_0, y_0) is the corresponding center of the circle and (P_x, P_y) are the actual coordinates. For general free-form contours the determination of contour error requires a more involved method. Cheng & Lee (2005) give a detailed algorithm for obtaining free-form contour errors. A simple method of approximating free-form contours is done by locally approximating the contour as a circle and if the axial errors are much smaller than the instantaneous radius of curvature, a contour error can be found by Eq. (2.10). It should be noted also that improving tracking performance does not necessarily guarantee good contouring performance (Koren & Lo, 1991).

2.5 REMARKS

This chapter presented various control algorithms that are designed with the intent of improving system performance. It was shown that virtually all of the presented control systems could be applied to the tracking control of robotic manipulators. The purpose of this chapter was to provide some insights into different control design methodologies and to outline key advantages and drawbacks of each. It was shown that if improved tracking performance is

desired then in general it requires some a priori knowledge of system dynamics. However, control laws that feedforward system dynamics behaviour can become difficult to implement.

Thus, some advanced control methods such as ILC offer ways to reduce the amount of required knowledge of the system while also improving performance. However, there may still exist issues regarding stability or ease of implementation. Progressing further, the notion of hybrid control systems was introduced. It was shown that combining two or more control laws by their desirable traits can significantly improve system dynamic response. To further the notion of hybrid control systems, the next chapter will contain formulation for a hybrid PD sliding mode control law. The main motivation for this research is to improve tracking and contouring performance over standard PD/PID control schemes while maintaining a similar level of ease of implementation by hybridizing with sliding mode control's discontinuous property to account for system nonlinearities.

Chapter 3 : HYBRID PD SLIDING MODE CONTROL

In this chapter the dynamics of a general robotic manipulator is presented. The formulation for PD control of the manipulator is presented and a general SMC law with included dynamic model uncertainties is formulated. A hybridization of PD and SMC is discussed along with the rationale and methodology behind it. Finally, the stability analysis of the proposed control law is conducted for the control of robotic manipulators with the Lyapunov method.

3.1 DYNAMIC MODEL

The dynamic equations of motion of a robotic manipulator embody the relationship between the torques/forces applied by the actuators and the resulting motion of the manipulator (Craig, 2005). The dynamics can be mathematically represented by the following nonlinear differential equation for an n -link rigid manipulator:

$$M(q)\ddot{q} + C(\dot{q}, q)\dot{q} + G(q) + D(t, \dot{q}, q) = \tau(t) \quad (3.1)$$

where:

- $q = q(t)$, $\dot{q} = \dot{q}(t)$, $\ddot{q} = \ddot{q}(t)$ are functions of time and are $n \times 1$ vectors representing the joint position, velocity, and acceleration, respectively.
- $M(q)$ is an $n \times n$ inertia matrix.
- $C(\dot{q}, q)$ is an $n \times n$ matrix containing the centrifugal-Coriolis terms.
- $G(q)$ is an $n \times 1$ vector of gravitational torques or forces.
- $D(t, \dot{q}, q)$ is an $n \times 1$ vector representing friction, uncertainty, disturbance terms.
- $\tau(t)$ is an $n \times 1$ vector of the joint torques or forces.

3.2 PD CONTROL FORMULATION

Defining the tracking errors of the system as follows:

$$\begin{cases} e(t) = q_d(t) - q(t) \\ \dot{e}(t) = \dot{q}_d(t) - \dot{q}(t) \\ \ddot{e}(t) = \ddot{q}_d(t) - \ddot{q}(t) \end{cases} \quad (3.2)$$

where $q_d(t)$, $\dot{q}_d(t)$, $\ddot{q}_d(t)$ are the vectors of desired joint position, velocity, and acceleration, respectively. By submitting Eqs. (3.2) into Eq. (3.1) the dynamic model can be rewritten in terms of tracking error as follows:

$$\begin{cases} M(q)\ddot{e} + C(\dot{q}, q)\dot{e} = \tau_d(t) - \tau(t) \\ \tau_d(t) = M(q)\ddot{q}_d + C(\dot{q}, q)\dot{q}_d + G(q) + D(t, \dot{q}, q) \end{cases} \quad (3.3)$$

where $\tau_d(t)$ is the desired input torque. For trajectory tracking control of rigid robotic manipulators a PD controller can be implemented as

$$\tau(t) = \tau_{PD}(t) = K_p e(t) + K_d \dot{e}(t) \quad (3.4)$$

where the positive definite matrices K_p and K_d are the proportional and derivative gains, respectively. It is seen that PD control is easy to implement as it does not require any previous knowledge of the specific system dynamics and requires only the tracking error of the system in order to compute the control torque. As such, it provides model-free linear feedback control for the system and is well-known to obtain acceptable tracking performance.

3.3 SMC FORMULATION

For standard sliding mode control, the tracking errors are redefined as follows:

$$\begin{cases} e(t) = q(t) - q_d(t) \\ \dot{e}(t) = \dot{q}(t) - \dot{q}_d(t) \\ \ddot{e}(t) = \ddot{q}(t) - \ddot{q}_d(t) \end{cases} \quad (3.5)$$

Note that this is essentially the opposite of the tracking error defined for standard PD/PID control. Using Eq. (3.5) as the definition of tracking errors, define the sliding surface and its time derivative as:

$$\begin{cases} s(t) = \dot{e}(t) + \lambda e(t) \\ \dot{s}(t) = \ddot{e}(t) + \lambda \dot{e}(t) \end{cases} \quad (3.6)$$

where $\lambda = \text{diag}[\lambda_1, \dots, \lambda_n]$ is a matrix containing the vector of slopes of the sliding surface, known as bandwidths, of the SMC and $\lambda_1, \dots, \lambda_n$ are strictly positive constants.

Define reference states as:

$$\begin{cases} \dot{q}_r(t) = \dot{q}(t) - s(t) = \dot{q}_d(t) - \lambda e(t) \\ \ddot{q}_r(t) = \ddot{q}(t) - \dot{s}(t) = \ddot{q}_d(t) - \lambda \dot{e}(t) \end{cases} \quad (3.7)$$

By submitting Eq. (3.5) to Eq. (3.7), into Eq. (3.1), the dynamic model can be rewritten in terms of the newly defined vector s :

$$M(q)\dot{s} + C(q, \dot{q})s = \tau(t) - M(q)\ddot{q}_r - C(q, \dot{q})\dot{q}_r - G(q) - D(t, q, \dot{q}) \quad (3.8)$$

Eq. (3.8) represents the system dynamics if the dynamic parameters are known exactly. Since in practice, the system dynamics parameters can never be defined exactly, the SMC law must take the deviations from the actual dynamics into account.

Define:

$$\Delta M(q) = \widehat{M}(q) - M(q) \quad (3.9)$$

$$\Delta C(q, \dot{q}) = \widehat{C}(q, \dot{q}) - C(q, \dot{q}) \quad (3.10)$$

$$\Delta G(q) = \widehat{G}(q) - G(q) \quad (3.11)$$

where $\widehat{M}(q)$, $\widehat{C}(q, \dot{q})$, and $\widehat{G}(q)$ are the estimations of the dynamic properties of the system $M(q)$, $C(q, \dot{q})$, and $G(q)$, respectively. Submitting Eq. (3.9) to Eq. (3.11) into Eq. (3.8), the dynamic equation is rewritten as:

$$M(q)\dot{s} + C(q, \dot{q})s = \tau - \hat{f} + \Delta f - D(t, q, \dot{q}) \quad (3.12)$$

where

$$\hat{f} = \widehat{M}(q)\ddot{q}_r + \widehat{C}(q, \dot{q})\dot{q}_r + \widehat{G}(q) \quad (3.13)$$

$$\Delta f = \Delta M(q)\ddot{q}_r + \Delta C(q, \dot{q})\dot{q}_r + \Delta G(q) \quad (3.14)$$

From this formulation, a standard SMC law can be written using the estimated system dynamic parameters in the form of Eq. (2.7) as:

$$\tau_{SMC}(t) = \widehat{M}(q)\ddot{q}_r + \widehat{C}(q, \dot{q})\dot{q}_r + \widehat{G}(q) - K_s \text{sgn}(s) \quad (3.15)$$

where the positive definite matrix K_s is a control parameter and $\text{sgn}()$ is the sign function defined as:

$$\text{sgn}(s) = \begin{cases} -1 & \text{if } s < 0 \\ 0 & \text{if } s = 0 \\ 1 & \text{if } s > 0 \end{cases} \quad (3.16)$$

An ideal sliding mode controller for a system of second-order is graphically represented in a block diagram, Figure 3.1.

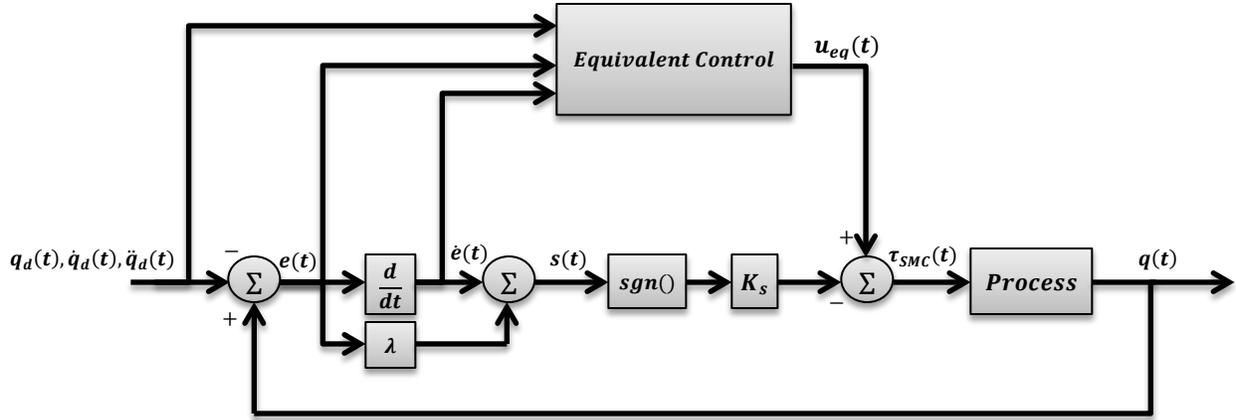


Figure 3.1: Block Diagram of ideal sliding mode control for a second order system.

Note that the block diagram shows an ideal SMC without the presence of model uncertainties or external loads and disturbances. The difference between the estimated systems dynamics and the actual system dynamics, Δf , as well as the friction and other disturbances, $D(t, q, \dot{q})$ are unaccounted for in the SMC law given in Eq. (3.12) and serve as sources of tracking error.

3.4 HYBRID PD SLIDING MODE CONTROL FORMULATION

From the previous sections, one can see that both PD control and SMC have their advantages and drawbacks. The problem, then, is combining these two different control methods in order to obtain a new control system. This section details the development of the new proposed hybrid control system.

3.4.1 Principle for Hybridization

There are two principles for the hybridization of two control systems: the complementary principle and the compatibility principle (Ouyang, 2005). The complementary principle states that two control systems can be integrated when the strengths of one of them correspond to and

will address the weaknesses of the other. The converse also applies with the strengths of the other one corresponding to the weaknesses of the first.

The compatibility principle states that the two control systems must be adjusted to minimize their side effects. Here lies the integration and stability problem. From this discussion one can see that the complementary principle gives a necessary condition for two control systems to be integrated while the compatibility principle gives a sufficient condition for the integration.

3.4.2 *Formulation*

The strengths and weaknesses of PD controllers and SMC can be summarized as follows. Standard PD controllers are very easy to implement since they use a model-free approach, yet they yield relatively low tracking performance due to their linear nature. On the other hand, SMC has high tracking performance but requires that the dynamics of the system be known a priori. The dynamic parameters of a system, however, may not be readily available or are difficult to obtain. As such, it is also sensitive to the unmodeled dynamics of the system. In order to solve the inherent drawbacks of both systems, the following PD sliding mode control (PD-SMC) law is proposed for the trajectory tracking control of rigid robotic manipulators:

$$\tau_H(t) = K_p e(t) + K_d \dot{e}(t) + K_s \text{sgn}(s) \quad (3.17)$$

$$s = \lambda e(t) + \dot{e}(t)$$

where for an n -link manipulator:

- $e(t)$ and $\dot{e}(t)$ are the $n \times 1$ vectors of position and velocity errors, respectively.
- s is the $n \times 1$ vector of sliding surfaces.

- K_p and K_d are the positive definite $n \times n$ diagonal matrices containing the proportional and derivative gains, respectively.
- K_s is a positive definite $n \times n$ diagonal matrix containing what are referred to as “SMC gains”.

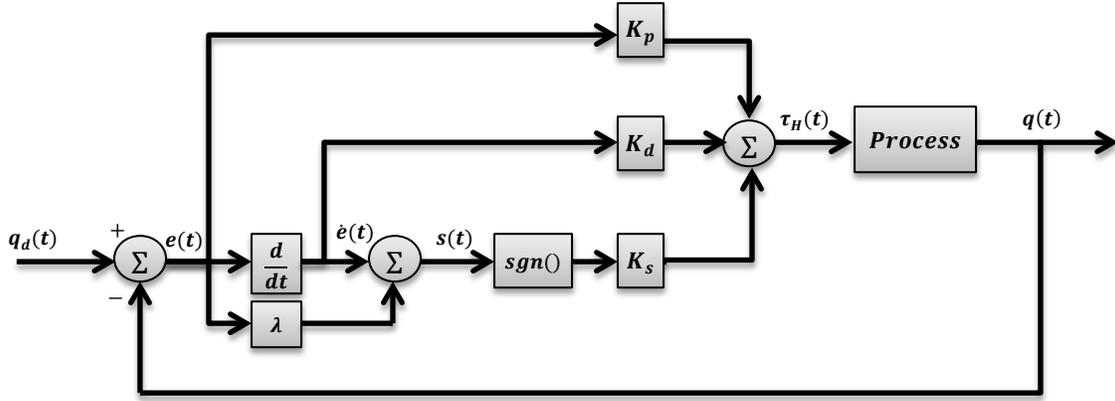


Figure 3.2: Block diagram of an ideal second order PD-SMC system.

Remark 1: The tracking errors for the proposed hybrid system are defined the same way as the tracking errors for PID control, Eq. (3.2). The definition of the sliding surface, s , is the same as Eq. (3.6), but also uses the tracking error definition of PID control, Eq. (3.2).

Remark 2: Comparing Eq. (3.17) to Eq. (3.15) and Eq. (2.7), as well as Figure 3.2 to Figure 3.1, it can be seen that the model-based equivalent control components in SMC is replaced by PD control in the proposed hybrid PD-SMC. Therefore, the proposed PD-SMC is model-free. Also, PD-SMC is a nonlinear feedback controller due to the switching term of the SMC.

Remark 3: The proposed PD-SMC law has the advantages of both PD control and SMC. First, it has the ease of implementation of PD control since it is only feedback control. This strength directly corresponds to the weakness of SMC that it is model-based. Second, it contains the nonlinear switching term contributed by SMC to provide improved tracking performance. This corresponds directly to the weakness of PD control that it only has a simple linear nature. Also, it is robust in that it handles uncertainties and external disturbances well as will be demonstrated through simulation verification.

Remark 4: Since in the PD-SMC law the strengths and weaknesses of one of the contributing control laws correspond to the weaknesses and strengths, respectively, of the other contributing control law, it satisfies the complementary principle for the hybridization of two control systems.

3.4.3 *Properties of the Dynamic Model*

A list of properties of a rigid robotic manipulator that are associated with Eq. (3.1) are used for the proposed hybrid PD-SMC law (Craig, 2005). These properties are used in the stability analysis in the following section. The properties are described as follows:

P1) The inertia matrix $M(q)$ is symmetric positive definite.

P2) The matrix $\frac{1}{2}\dot{M}(q) - C(q, \dot{q})$ is a skew symmetric matrix.

P3) The inertial and the centrifugal-Coriolis matrices satisfy the following relationship:

$$\dot{M}(q) = C(q, \dot{q}) + C^T(q, \dot{q}) \quad (3.18)$$

P4) $M(q)$, $C(\dot{q}, q)$, $G(q)$, and $D(t, \dot{q}, q)$ are all bounded for bounded inputs.

3.5 STABILITY ANALYSIS

3.5.1 Notations and Theorem

To facilitate the stability analysis, the following notations are introduced.

Let $\lambda_m(M)$ and $\lambda_M(M)$ represent the smallest and largest eigenvalues of a positive definite matrix, M . If a square matrix M is positive definite, then it is denoted as $M > 0$. If a square matrix $M - N$ is positive definite, then it is denoted as $M - N > 0$.

For positive definite matrices, the following properties will be used (Boyd & Vandenberghe, 2004).

P5) If $M > 0$, then $M^{-1} > 0$.

P6) If $M \geq N > 0$, then $N^{-1} \geq M^{-1} > 0$.

P7) If $M > 0$ and $\lambda > 0$ is a real number, then $\lambda M > 0$.

P8) If $M > 0$ and $N > 0$, then $M + N > 0$, $MNM > 0$, and $NMN > 0$. Also if $MN = NM$, then $MN > 0$.

Theorem: For a rigid robotic manipulator described in Eq. (3.1), if the proposed hybrid PD-SMC law in Eq. (3.17) is applied to control a trajectory tracking of the robotic manipulator, and the following conditions in Eqs. (3.19) are satisfied, then the controlled robotic manipulator is globally asymptotically stable for the trajectory tracking and the tracking errors converge to zeros.

$$\left\{ \begin{array}{l} \lambda > 0 \\ K_s > \|\tau_d\| \\ K + C^T > 0 \\ \lambda_m(K_d) > \lambda \cdot \lambda_M(M) \\ \lambda_m(K_p) > \lambda^2 \cdot \lambda_M(M) \\ \lambda_m(K_d) > \frac{1}{2} \cdot \lambda_M(K + C^T) \\ \lambda_m(K_d - \lambda M) > \frac{\lambda}{2} \cdot \lambda_M(K + C^T) \end{array} \right. \quad (3.19)$$

where K is a user-selected constant positive definite matrix.

3.5.2 Proposition

Let us introduce the following.

Proposition: Assume a matrix Q is a symmetric matrix expressed as:

$$Q = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \quad (3.20)$$

Let S be the Schur complement (Bristow, Tharayil, & Alleyne, 2006) of matrix A in Q , that is,

$$S = C - B^T A^{-1} B \quad (3.21)$$

Then the matrix Q is positive definite if and only if the matrices A and S are both positive definite. This means that if $A > 0$ and $S > 0$, then $Q > 0$.

Proof: See reference (Boyd & Vandenberghe, 2004).

To prove the stability of the proposed hybrid PD-SMC law, the following matrix Q must first be proved to be positive definite.

$$Q = \begin{bmatrix} K_p & \lambda M \\ (\lambda M)^T & M \end{bmatrix} \quad (3.22)$$

Proof:

As PD control gains are symmetric diagonal matrices with positive constant elements, K_p is a positive definite matrix $K_p > 0$. From property P1, we know that the matrix M is symmetric positive definite, i.e. $M = M^T$ and $M > 0$. Therefore, the matrix Q is a symmetric matrix.

From conditions Eqs. (3.19), we have:

$$\lambda_m(K_p) > \lambda^2 \cdot \lambda_M(M) > 0 \quad (3.23)$$

As K_p and M are positive definite, from Eq. (3.23) we can conclude:

$$K_p - \lambda^2 M > 0 \quad (3.24)$$

As $K_p > 0$ and $M > 0$, then according to properties P5 – P7, we obtain:

$$M^{-1} > \lambda^2 K_p^{-1} > 0 \quad (3.25)$$

According to the definition of a positive definite matrix, then Eq. (3.25) can be written as:

$$M^{-1} - \lambda^2 K_p^{-1} > 0 \quad (3.26)$$

Furthermore, based on Eq. (3.26) and $M > 0$, then according to property P8, we have:

$$M(M^{-1} - \lambda^2 K_p^{-1})M > 0 \quad (3.27)$$

According to the property P1 and reorganizing Eq. (3.27) we have:

$$S = M - (\lambda M)^T K_p^{-1} (\lambda M) > 0 \quad (3.28)$$

Therefore, according to the Proposition and Eq. (3.21), we have proven that the matrix Q in Eq. (3.22) is symmetric and positive definite.

3.5.3 Proof of Stability

For the proposed hybrid PD-SMC law, we define the following Lyapunov function:

$$\begin{aligned} V &= \frac{1}{2} (e^T \quad \dot{e}^T) Q \begin{pmatrix} e \\ \dot{e} \end{pmatrix} + \frac{\lambda}{2} e^T (K + K_d) e \\ &= \frac{1}{2} (e^T \quad \dot{e}^T) \begin{bmatrix} K_p & \lambda M \\ (\lambda M)^T & M \end{bmatrix} \begin{pmatrix} e \\ \dot{e} \end{pmatrix} + \frac{\lambda}{2} e^T (K + K_d) e \end{aligned} \quad (3.29)$$

Since Q , K , and K_d are symmetric positive definite matrices, the Lyapunov function in Eq. (3.29) is a positive definite function for all e and \dot{e} :

$$V(e(t), \dot{e}(t)) > 0 \quad (3.30)$$

The time derivative of the Lyapunov function in Eq. (3.29) along the tracking errors can be written as:

$$\begin{aligned} \dot{V} &= (e^T \quad \dot{e}^T) \begin{bmatrix} K_p & \lambda M \\ (\lambda M)^T & M \end{bmatrix} \begin{pmatrix} \dot{e} \\ \ddot{e} \end{pmatrix} \\ &+ \frac{1}{2} (e^T \quad \dot{e}^T) \begin{bmatrix} 0 & \lambda \dot{M} \\ (\lambda \dot{M})^T & \dot{M} \end{bmatrix} \begin{pmatrix} e \\ \dot{e} \end{pmatrix} + \lambda e^T (K + K_d) \dot{e} \\ &= (e^T \quad \dot{e}^T) \begin{pmatrix} K_p \dot{e} + \frac{\lambda \dot{M}}{2} \dot{e} + \lambda M \ddot{e} \\ \frac{\lambda \dot{M}}{2} e + \lambda M \dot{e} + \frac{\dot{M}}{2} \dot{e} + M \ddot{e} \end{pmatrix} + \lambda e^T (K + K_d) \dot{e} \end{aligned} \quad (3.31)$$

From Eq. (3.3), Eq. (3.6), and Eq. (3.17) we have:

$$M \ddot{e} = \tau_d - C \dot{e} - K_p e - K_d \dot{e} - K_s \text{sgn}(\dot{e} + \lambda e) \quad (3.32)$$

According to property P2, we have:

$$x^T \left(\frac{\dot{M}}{2} - C \right) x = 0 \quad (3.33)$$

From the properties of skew symmetric matrices, Eq. (3.33) holds true for all x . Applying Eq. (3.33) and Eq. (3.32) into Eq. (3.31) and using property P3 we obtain the following:

$$\begin{aligned}\dot{V} = & -\lambda e^T K_p e - \dot{e}^T (K_d - \lambda M) \dot{e} + \lambda \dot{e}^T (K + C^T) e \\ & + (\lambda e^T + \dot{e}^T) (\tau_d - K_s \text{sgn}(e + \lambda e))\end{aligned}\quad (3.34)$$

According to the conditions in Eq. (3.19), we have $K + C^T > 0$, then the following inequality holds:

$$0 < K + C^T < \lambda_M (K + C^T) \quad (3.35)$$

From Eq. (3.35), we have:

$$\dot{e}^T (K + C^T) e \leq e^T \frac{\lambda_M (K + C^T)}{2} I e + \dot{e}^T \frac{\lambda_M (K + C^T)}{2} I \dot{e} \quad (3.36)$$

Submitting Eq. (3.36) into Eq. (3.34) and simplifying, we obtain the following:

$$\begin{aligned}\dot{V} \leq & -\lambda e^T \left(K_p - \frac{1}{2} \lambda_M (K + C^T) I \right) e - \dot{e}^T \left(K_d - \lambda M - \frac{\lambda}{2} \lambda_M (K + C^T) I \right) \dot{e} \\ & + (\lambda e^T + \dot{e}^T) (\tau_d - K_s \text{sgn}(e + \lambda e))\end{aligned}\quad (3.37)$$

According to the conditions in Eq. (3.19) we have:

$$\begin{cases} K_p - \frac{1}{2} \lambda_M (K + C^T) I > 0 \\ K_d - \lambda M - \frac{\lambda}{2} \lambda_M (K + C^T) I > 0 \end{cases} \quad (3.38)$$

Therefore, we can prove that:

$$\begin{cases} -\lambda e^T \left(K_p - \frac{1}{2} \lambda_M (K + C^T) I \right) e \leq 0 \\ -\dot{e}^T \left(K_d - \lambda M - \frac{\lambda}{2} \lambda_M (K + C^T) I \right) \dot{e} \leq 0 \end{cases} \quad (3.39)$$

Also, according to condition Eq. (3.19) and the property P4, we have:

$$\begin{aligned}
(\lambda e^T + \dot{e}^T)K_s \text{sgn}(\dot{e} + \lambda e) &= |\lambda e^T + \dot{e}^T|K_s \\
&\geq |\lambda e^T + \dot{e}^T| \|\tau_d\| \\
&\geq (\lambda e^T + \dot{e}^T)\tau_d
\end{aligned} \tag{3.40}$$

From Eq. (3.40), we have:

$$(\lambda e^T + \dot{e}^T)(\tau_d - K_s \text{sgn}(\dot{e} + \lambda e)) \leq 0 \tag{3.41}$$

Taking Eq. (3.41) and Eq.(3.39) and submitting them into Eq. (3.37), we finally conclude:

$$\dot{V} \leq 0 \tag{3.42}$$

The only instance that $\dot{V} = 0$ is when $e = \dot{e} = 0$. Since the Lyapunov function V is positive definite and \dot{V} is negative definite, the robot manipulator represented in Eq. (3.1) and controlled by the proposed hybrid PD-SMC law Eq. (3.17) is globally asymptotically stable based on the Lyapunov method, and the tracking error and its derivative are zeros. This means that:

$$\begin{cases} \lim_{t \rightarrow \infty} e = 0 \\ \lim_{t \rightarrow \infty} \dot{e} = 0 \end{cases} \tag{3.43}$$

3.6 REMARKS

It should be noted that the switching term in the hybrid PD-SMC law Eq. (3.17) is discontinuous and could introduce chatter in the system. In order to avoid this, a method of

smoothing the controlled input torque will be used by introducing a boundary layer (ϕ) along with a saturation function (Slotine & Li, 1991). The modified control law is:

$$\tau_H(t) = K_p e(t) + K_d \dot{e}(t) + K_s \text{sat}\left(\frac{s}{\phi}\right) \quad (3.44)$$

The saturation function is defined as:

$$\text{sat}\left(\frac{s}{\phi}\right) = \begin{cases} \frac{s}{\phi} & \text{if } \left|\frac{s}{\phi}\right| \leq 1 \\ \text{sgn}\left(\frac{s}{\phi}\right) & \text{otherwise} \end{cases} \quad (3.45)$$

Chapter 4 : SIMULATION AND RESULTS

In this chapter, a number of simulations for trajectory tracking control of robot manipulators will be performed based on the proposed hybrid PD-SMC law. To do this, two virtual robotic manipulators will be used. The first manipulator is a three-DOF serial planar manipulator with a 3R configuration. The second manipulator is a two-DOF closed-chain planar manipulator in a parallel 5R configuration. In addition to establishing the mechanisms required for simulation, detailed trajectory planning is performed to ensure smooth motions in all active joints.

4.1 SIMULATION SETUP

4.1.1 *Serial Manipulator*

The serial robotic manipulator used for the simulation consists of $n = 3$ links of length L_i that are actuated about three revolute joints where the subscript i denotes the link or joint number. The motion of the manipulator is planar and the links are rigid with centers of mass at a distance r_i from the preceding joint along its length. The angle of each link relative to the preceding one is denoted by q_i . The end-effector Cartesian coordinates are denoted (x_{ee}, y_{ee}) while the pose of the end-effector, q , is denoted without a subscript and is the sum of the joint angles of the manipulator. A diagram of the basic 3R configuration is shown in Figure 4.1 and the specific structural parameters used for the manipulator in the simulations are listed in Table 4.1.

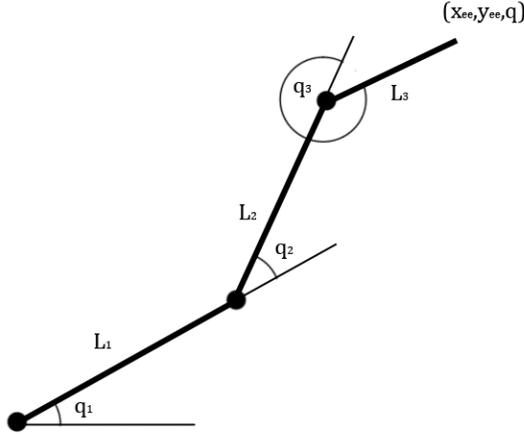


Figure 4.1: Serial 3R Robotic Manipulator

Table 4.1: Structural Parameters of the 3R Serial Manipulator

Link i	Mass m_i (kg)	Length L_i (m)	Distance to Mass Center r_i (m)	Inertia I_i (kgm ²)
1	1.00	0.50	0.25	0.10
2	1.00	0.50	0.25	0.10
3	0.50	0.30	0.15	0.05

Also integrated into the dynamic model of the 3R manipulator are the moment of inertia and viscous damping of a servomotor at each controlled joint. The integrated model of the manipulator then is represented by:

$$\bar{M}(q)\ddot{q} + C(q, \dot{q})\dot{q} + B\dot{q} + G(q) = \tau(t) \quad (4.1)$$

where $\bar{M} = M + J$, J is a 3×3 matrix containing servomotor inertias and B is a 3×3 matrix containing the servomotor viscous damping coefficients. Each motor has $J_i = 0.05 \text{ kgm}^2$ and $B_i = 0.07 \text{ Nms}$. The detailed dynamic parameters for the 3R manipulator are located in the Appendices.

4.1.2 Parallel Manipulator

The parallel manipulator used for the simulation consists of $n = 5$ links of length L_i with two degrees of freedom. The scheme of the manipulator and its workspace are shown in Figure 4.2. The active angles of the manipulator are q_1 and q_2 , actuated about points O_1 and O_2 . The numbering conventions for the link masses, joints, centers of mass, and moments of inertia are the same as those for the serial manipulator. Also like the serial manipulator, the links are assumed to be rigid and the motion of the manipulator is planar. The specific structural parameters for the parallel manipulator used in the simulations is are listed in Table 4.2. The base link, whose joints are located at O_1 and O_2 , is denoted as Link 5.

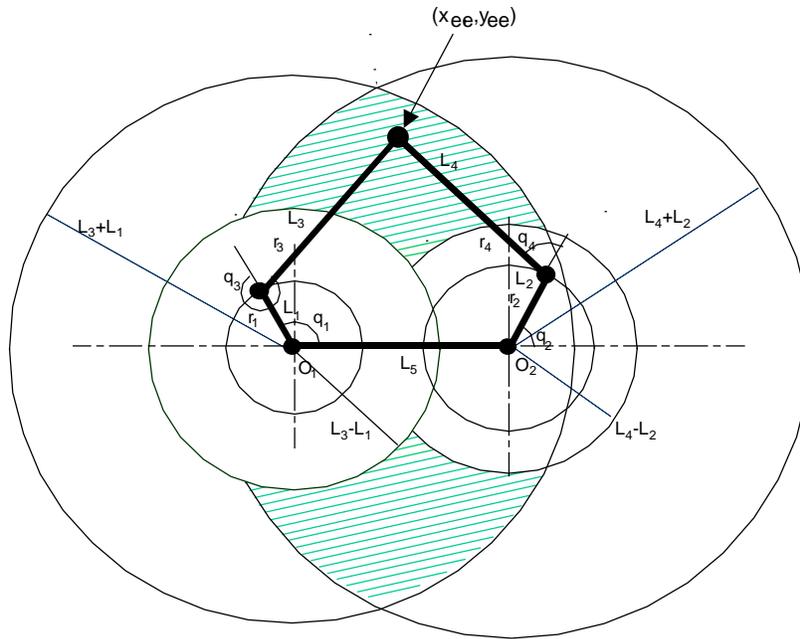


Figure 4.2: Parallel 5R Robotic Manipulator

Because this is a two-DOF mechanism, q_3 and q_4 are written as functions of q_1 and q_2 . The dynamic model of the mechanism is derived using the reduced model method given by (Ghorbel, 1995) and is derived as:

$$\begin{cases} \bar{M}(q')\ddot{q} + C(q', \dot{q}')\dot{q} + B\dot{q} + G(q') = \tau(t) \\ \dot{q}' = \rho(q')\dot{q} \\ q' = \sigma(q) \end{cases} \quad (4.2)$$

where

$$q = [q_1 \quad q_2]^T, \quad q' = [q_1 \quad q_2 \quad q_3 \quad q_4]^T$$

$$\dot{q} = [\dot{q}_1 \quad \dot{q}_2]^T, \quad \dot{q}' = [\dot{q}_1 \quad \dot{q}_2 \quad \dot{q}_3 \quad \dot{q}_4]^T$$

and the dynamic parameter coefficients are defined the same way as for the serial manipulator except are 2×2 instead of 3×3 because of the change in the number of degrees of freedom. Each motor has $J_i = 0.05 \text{ kgm}^2$ and $B_i = 0.05 \text{ Nms}$. The definitions of $\rho(q')$, $\sigma(q)$, as well as the detailed dynamic parameters of the 5R manipulator are given in the Appendices.

Table 4.2: Structural Parameters of the 5R Parallel Manipulator

Link i	Mass $m_i \text{ (kg)}$	Length $L_i \text{ (m)}$	Distance to Mass Center $r_i \text{ (m)}$	Inertia $I_i \text{ (} \times 10^{-2} \text{ kgm}^2 \text{)}$
1	0.91	0.080	0.006	0.847
2	0.28	0.100	0.028	0.630
3	0.38	0.250	0.125	4.002
4	0.38	0.250	0.125	4.002
5	—	0.250	—	—

These link lengths ensure that the input links of the parallel manipulator have full rotatability (Ting, 1989).

4.2 TRAJECTORY PLANNING FOR SIMULATIONS

The desired trajectories for each link of the mechanisms are composed of a number of path segments. The mechanisms exhibit stop-and-go motion for each segment. In order to ensure smooth trajectories for each joint, a fifth order polynomial is used to parametrically establish

segment's position, velocity, and acceleration as a function of time (Craig, 2005). A constraint is applied that ensures the initial and final velocities and accelerations for each path segment are zero. The fifth-order polynomial is given as:

$$s_c(t) = 6 \left(\frac{t}{T}\right)^5 - 15 \left(\frac{t}{T}\right)^4 + 10 \left(\frac{t}{T}\right)^3 \quad (4.3)$$

The first and second derivative are used for define the desired velocities and accelerations, respectively, of each segment and are given as:

$$\begin{cases} \dot{s}_c(t) = \frac{1}{T} \left[30 \left(\frac{t}{T}\right)^4 - 60 \left(\frac{t}{T}\right)^3 + 30 \left(\frac{t}{T}\right)^2 \right] \\ \ddot{s}_c(t) = \frac{1}{T^2} \left[120 \left(\frac{t}{T}\right)^3 - 180 \left(\frac{t}{T}\right)^2 + 60 \left(\frac{t}{T}\right) \right] \end{cases} \quad (4.4)$$

where t is the amount of time elapsed into the current segment and T is the total duration of the current path segment. A sampling rate of 500[Hz] was sufficient for contours for the serial manipulator, while a higher sampling rate of 5000 [Hz] was used for the parallel manipulator contours in order to smooth the control torque.

4.2.1 *Linear Contours*

Using the parameters defined in Eqs. (4.3) and (4.4), a contour composed of linear segments is defined on the end-effector level. The end-effector Cartesian coordinates are defined by the vector P , where $P = (x_{ee}, y_{ee}, q)$ for the serial manipulator and q is the pose angle. For the parallel manipulator, $P = (x_{ee}, y_{ee})$. Each segment is defined as follows:

$$P_d(t) = (P_f - P_i)s_c(t) \quad (4.5)$$

where $P_d(t)$ is the desired contour segment vector as a function of time, and P_i and P_f are the initial and final values of the contour segment vector, respectively. A series of linear translation segments of the end-effector's Cartesian coordinates comprise the diamond and zigzag contours of the manipulators. Given the desired trajectory of the end-effector, the desired trajectories in the joint space are calculated using the inverse kinematics in the Appendices. The endpoints for each segment of the zigzag and diamond contours for the serial manipulator in the task space are listed in Table 4.3. It should be noted that each segment for the linear contours has a duration of 2 seconds. The joint space positions and velocities for the serial manipulator are listed in Table 4.4. For the parallel manipulator, the contour endpoint information is listed in Table 4.5 for the task space and in Table 4.6 for the joint space. Figure 4.3 to Figure 4.14 show the joint space and task space positions and velocities for both manipulators.

Table 4.3: Linear Contour Segment End-effector Endpoints for Serial Manipulator

	Segment 1		Segment 2		Segment 3		Segment 4	
	P_i	P_f	P_i	P_f	P_i	P_f	P_i	P_f
Zigzag								
$x_{ee}(m)$	-0.50	-0.25	-0.25	0	0	0.25	0.25	0.50
$y_{ee}(m)$	0	0.50	0.50	0	0	0.50	0.50	0
$q(rad)$	-0.75π	$-\pi$	$-\pi$	-0.50π	-0.50π	0	0	-0.25π
Diamond								
$x_{ee}(m)$	-0.15	0	0	0.15	0.15	0	0	-0.15
$y_{ee}(m)$	0	-0.70	-0.70	0	0	0.70	0.70	0
$q(rad)$	0	-0.50π	-0.50π	$-\pi$	$-\pi$	-1.5π	-1.5π	-2.0π

Table 4.4: Linear Contour Segment Joint Position Endpoints for Serial Manipulator

	Segment 1		Segment 2		Segment 3		Segment 4	
	P_i	P_f	P_i	P_f	P_i	P_f	P_i	P_f
Zigzag								
$q_1(rad)$	7.585	6.710	6.710	6.588	6.588	6.909	6.909	5.713
$q_2(rad)$	-3.873	-4.195	-4.195	-3.751	-3.751	-4.195	-4.195	-3.873
$q_3(rad)$	-6.069	-5.657	-5.657	-4.408	-4.408	-2.715	-2.715	-2.626
Diamond								
$q_1(rad)$	2.038	3.553	3.553	5.179	5.179	6.695	6.695	8.321
$q_2(rad)$	2.208	2.319	2.319	2.208	2.208	2.319	2.319	2.208
$q_3(rad)$	-4.246	-7.442	-7.442	-10.53	-10.53	-13.73	-13.73	-16.81

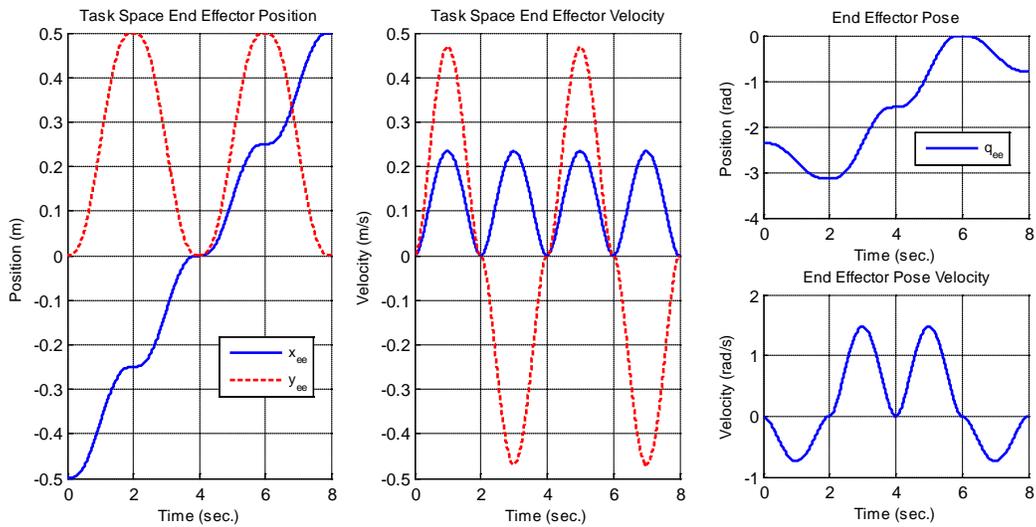


Figure 4.3: Zigzag Contour Task Space Positions and Velocities for Serial Manipulator

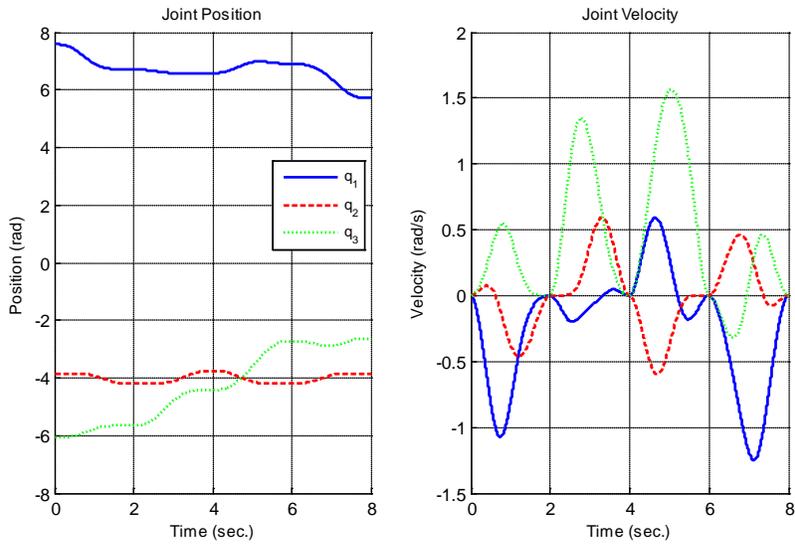


Figure 4.4: Zigzag Contour Joint Positions and Velocities for Serial Manipulator

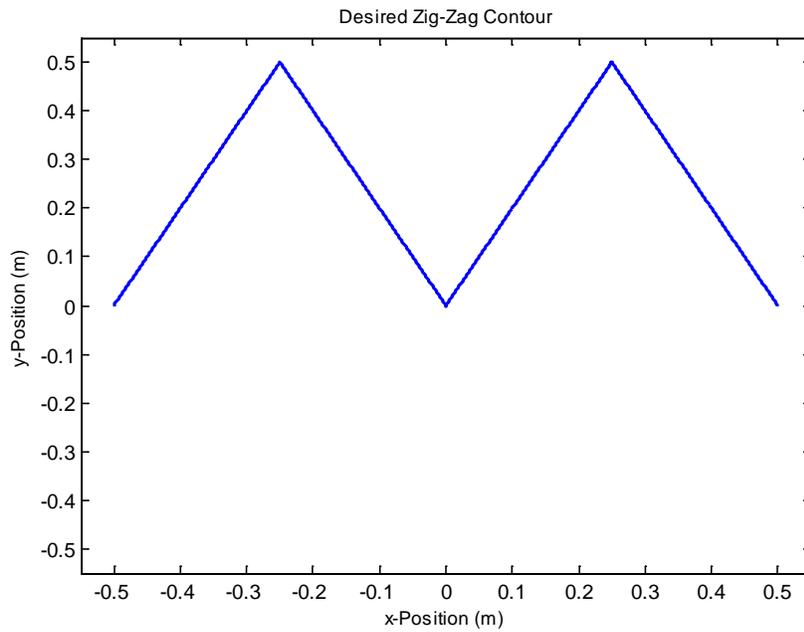


Figure 4.5: Desired Zigzag End-effector Contour for Serial Manipulator

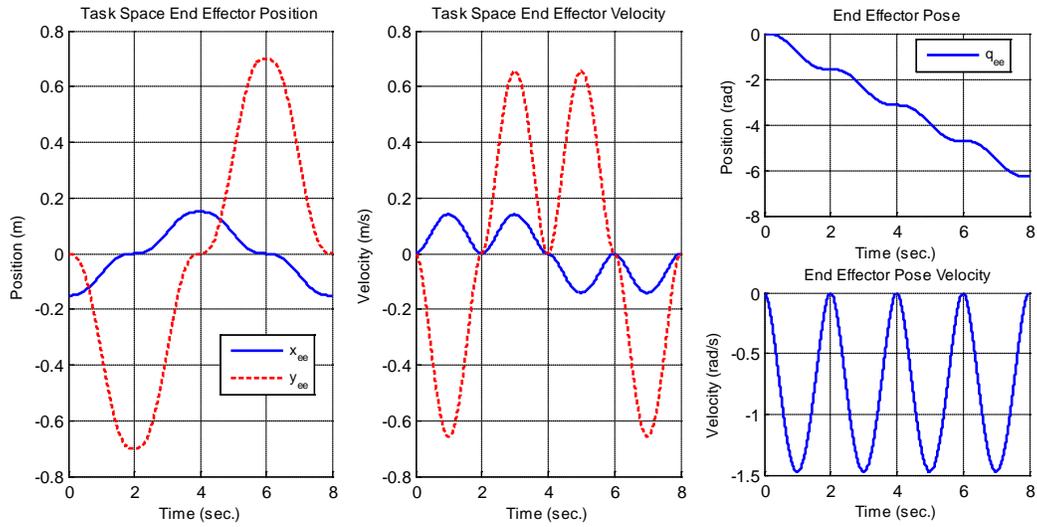


Figure 4.6: Diamond Contour Task Space Positions and Velocities for Serial Manipulator

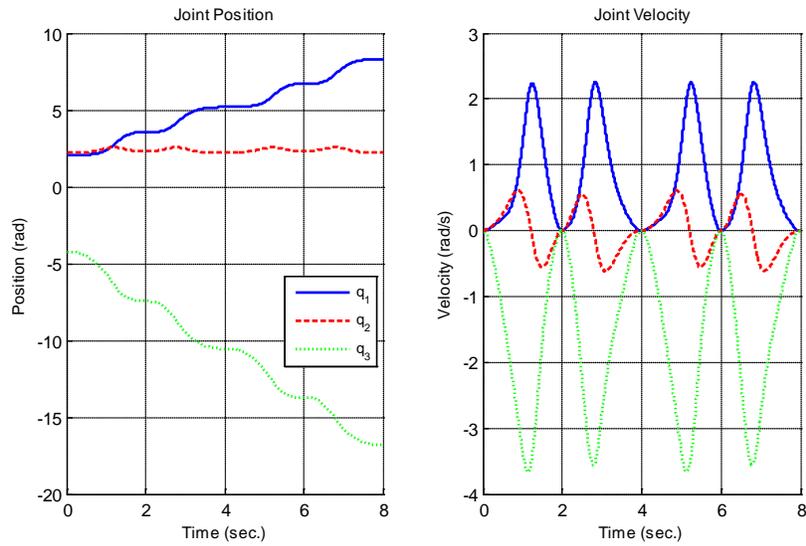


Figure 4.7: Diamond Contour Joint Positions and Velocities for Serial Manipulator

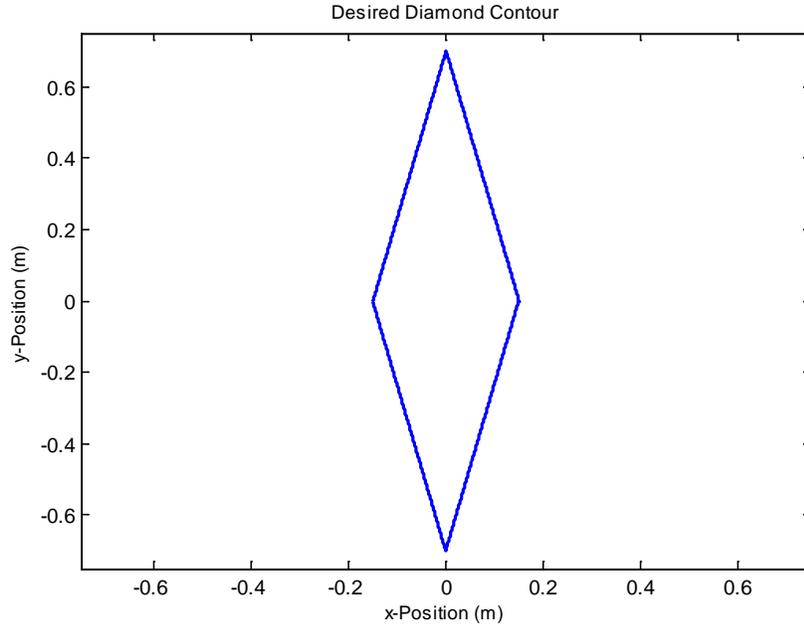


Figure 4.8: Desired Diamond End-effector Contour for Serial Manipulator

Table 4.5: Linear Contour Segment End-effector Endpoints for Parallel Manipulator

	Segment 1		Segment 2		Segment 3		Segment 4	
	P_i	P_f	P_i	P_f	P_i	P_f	P_i	P_f
Zigzag								
$x_{ee}(m)$	0.10	0.13	0.13	0.16	0.16	0.19	0.19	0.22
$y_{ee}(m)$	0.18	0.22	0.22	0.18	0.18	0.22	0.22	0.18
Diamond								
$x_{ee}(m)$	0.10	0.16	0.16	0.22	0.22	0.16	0.16	0.10
$y_{ee}(m)$	0.20	0.26	0.26	0.20	0.20	0.14	0.14	0.20

Table 4.6: Linear Contour Segment Joint Position Endpoints for Parallel Manipulator

	Segment 1		Segment 2		Segment 3		Segment 4	
	P_i	P_f	P_i	P_f	P_i	P_f	P_i	P_f
Zigzag								
$q_1(rad)$	3.063	2.381	2.381	2.366	2.366	1.772	1.772	1.682
$q_2(rad)$	0.746	0.706	0.706	0.161	0.161	0.255	0.255	-0.389
$q_3(rad)$	-2.295	-1.661	-1.661	-1.847	-1.847	-1.170	-1.170	-1.269
$q_4(rad)$	1.931	1.766	1.766	2.265	2.265	1.993	1.993	2.472
Diamond								
$q_1(rad)$	2.849	1.727	1.727	1.562	1.562	2.616	2.616	2.849
$q_2(rad)$	0.845	0.768	0.768	-0.142	-0.142	-0.269	-0.269	0.845
$q_3(rad)$	-2.063	-0.918	-0.918	-1.062	-1.062	-2.205	-2.205	-2.063
$q_4(rad)$	1.772	1.507	1.507	2.255	2.255	2.681	2.681	1.772

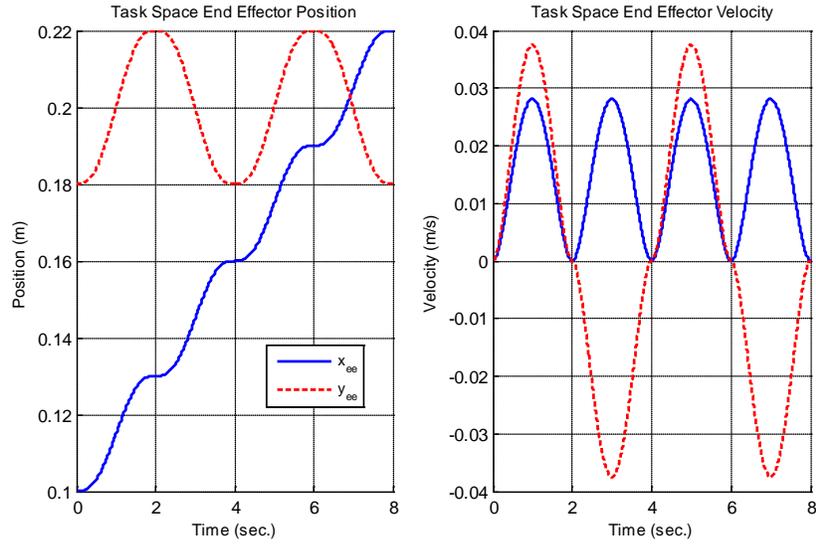


Figure 4.9: Zigzag Contour Task Space Positions and Velocities for Parallel Manipulator

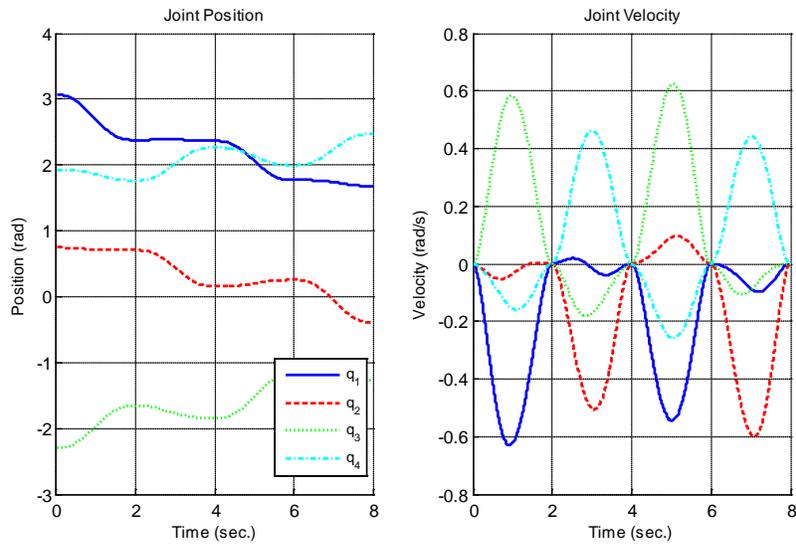


Figure 4.10: Zigzag Contour Joint Positions and Velocities for Parallel Manipulator

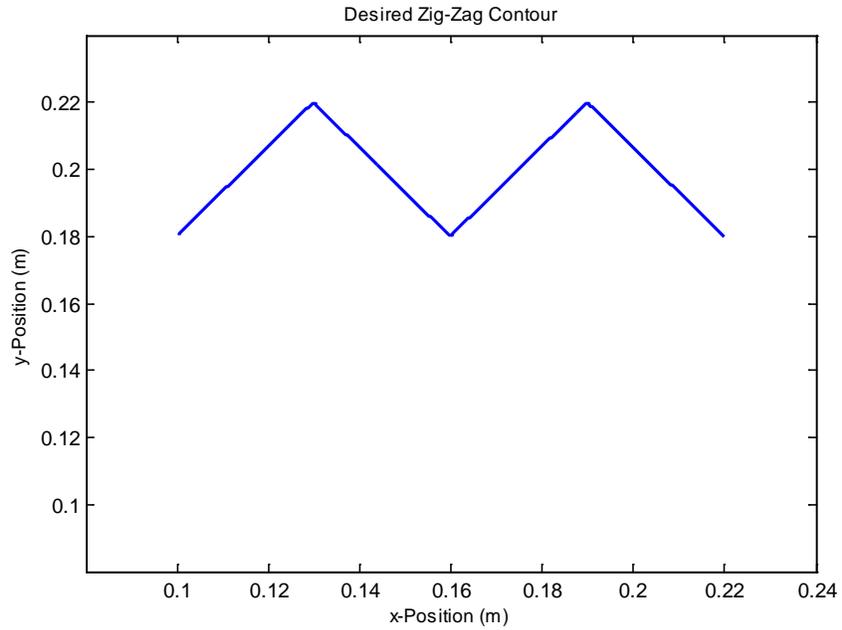


Figure 4.11: Desired Zigzag End-effector Contour for Parallel Manipulator

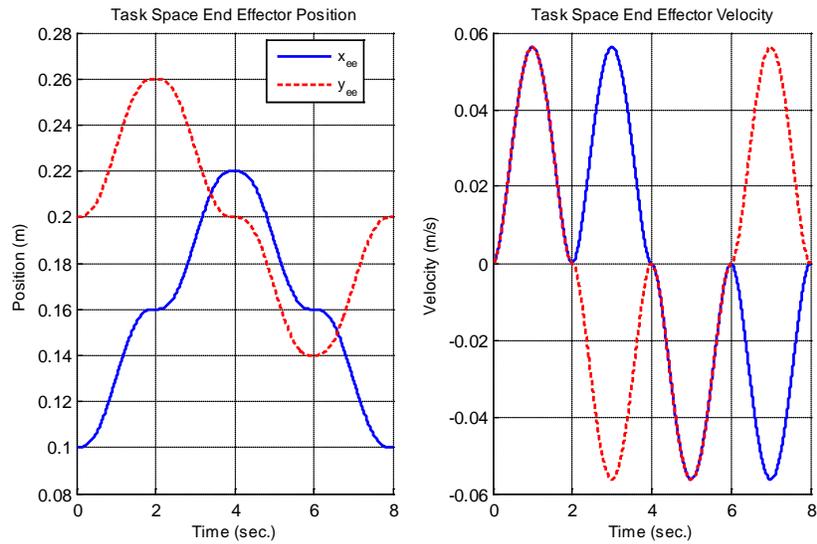


Figure 4.12: Diamond Contour Task Space Positions and Velocities for Parallel Manipulator

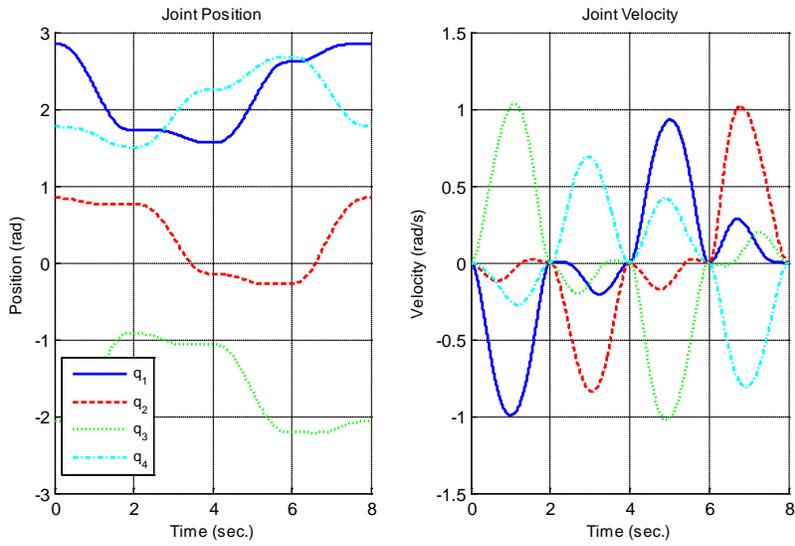


Figure 4.13: Diamond Contour Joint Positions and Velocities for Parallel Manipulator

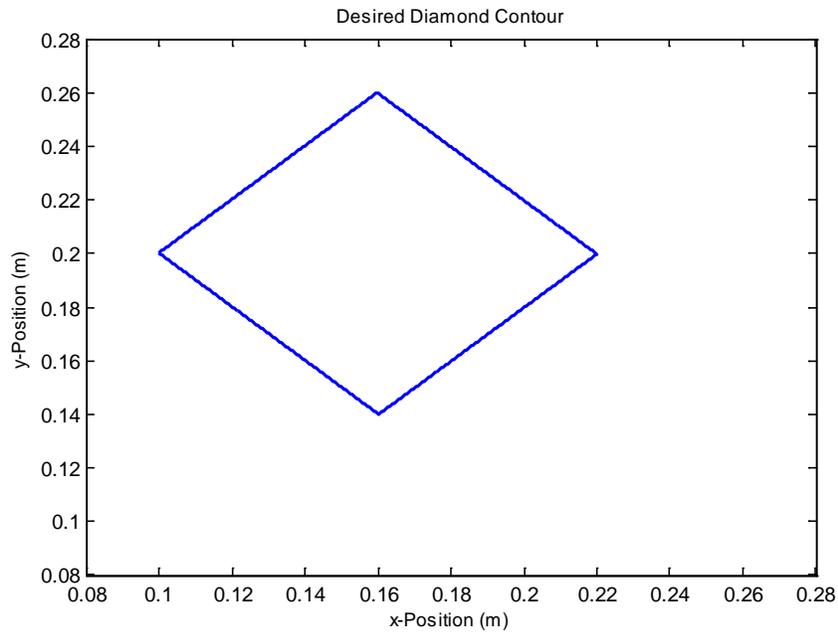


Figure 4.14: Desired Diamond End-effector Contour for Parallel Manipulator

4.2.2 *Nonlinear Contours*

Nonlinear contours are constructed parametrically as a function of time using an angular parameter defined as:

$$\theta_d(t) = (\theta_f - \theta_i)s_c(t) \quad (4.6)$$

where θ_i and θ_f are the initial and final values of this parameter, respectively. For both the serial and parallel manipulators, circular and elliptical contours are used for the simulation composed of a single movement segment. For each contour, $\theta_i = 0[rad]$ and $\theta_f = 2\pi [rad]$, indicating a single end-effector revolution with smooth joint motion, as ensured by the definition of $s(t)$.

A circular end-effector contour is defined parametrically as a function of time by:

$$\begin{cases} x_{ee}(t) = x_0 + R \cos(\theta_d(t)) \\ y_{ee}(t) = y_0 + R \sin(\theta_d(t)) \end{cases} \quad (4.7)$$

where the coordinates (x_0, y_0) are the coordinates of the center of the circle and R represents the radius.

The parametric representation of an elliptical contour as a function of time and centered at (x_0, y_0) is similar to that of a circular contour:

$$\begin{cases} x_{ee}(t) = x_0 + R_x \cos(\theta_d(t)) \\ y_{ee}(t) = y_0 + R_y \sin(\theta_d(t)) \end{cases} \quad (4.8)$$

where R_x is the radius of the ellipse along the x -axis and R_y is the radius of the ellipse along the y -axis. Eq. (4.8) reduces to the parametric equation of a circle when $R_x = R_y = R$.

The circular contour parameters for the serial manipulator are shown in Table 4.7 while the elliptical contour parameters are shown in Table 4.9. The initial and final coordinates and poses for serial manipulator circular and elliptical contours are given in Table 4.8, where the

pose is determined in the same manner as Eq. (4.5). The contour information for the serial manipulator in both the task and joint spaces are shown graphically in Figure 4.15 to Figure 4.20.

Table 4.7: Circular Contour Parameters for Serial Manipulator

x_0 (m)	y_0 (m)	R (m)	Duration (sec)
0	0.1	0.4	5

Table 4.8: Circular Contour Initial and Final Positions for Serial Manipulator

	Initial Value	Final Value
x_{ee} (m)	0.4	0.4
y_{ee} (m)	0.1	0.1
q (rad)	0	0

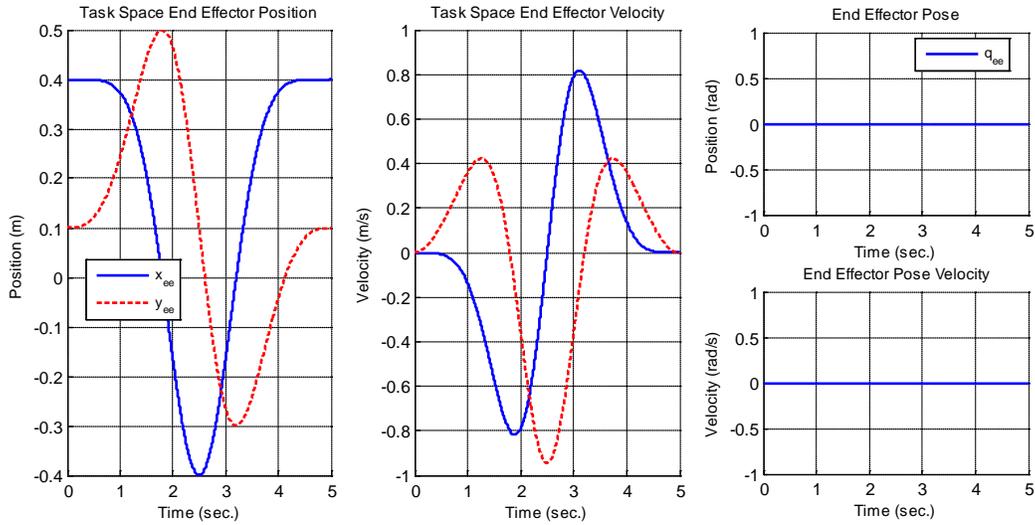


Figure 4.15: Circular Contour Task Space Positions and Velocities for Serial Manipulator

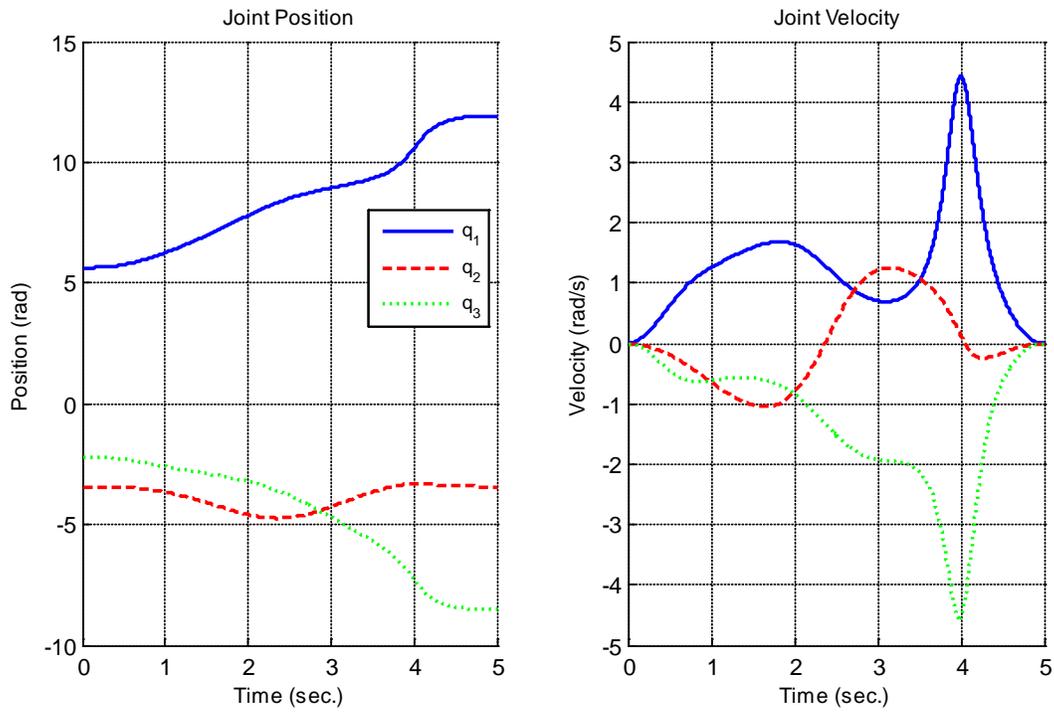


Figure 4.16: Circular Contour Joint Positions and Velocities for Serial Manipulator

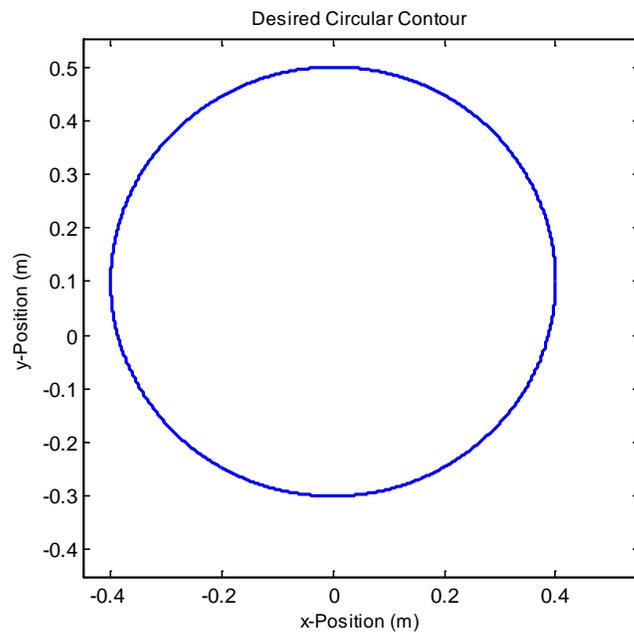


Figure 4.17: Desired Circular End-effector Contour for Serial Manipulator

Table 4.9: Elliptical Contour Parameters for Serial Manipulator

x_0 (m)	y_0 (m)	R_x (m)	R_y (m)	Duration (sec)
0	0	0.2	0.5	8

Table 4.10: Elliptical Contour Initial and Final Positions for Serial Manipulator

	Initial Value	Final Value
x_{ee} (m)	0.2	0.2
y_{ee} (m)	0	0
q (rad)	$-\pi$	π

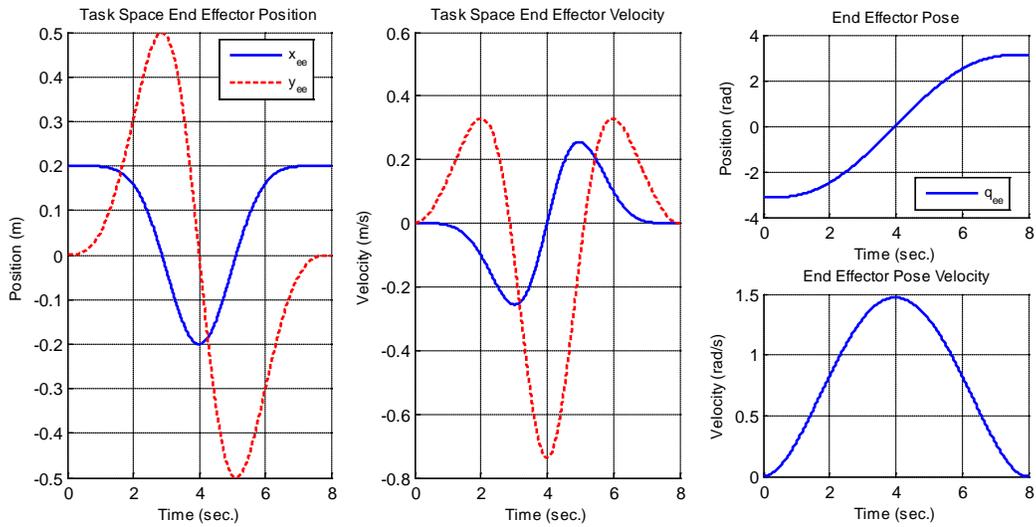


Figure 4.18: Elliptical Contour Task Space Positions and Velocities for Serial Manipulator

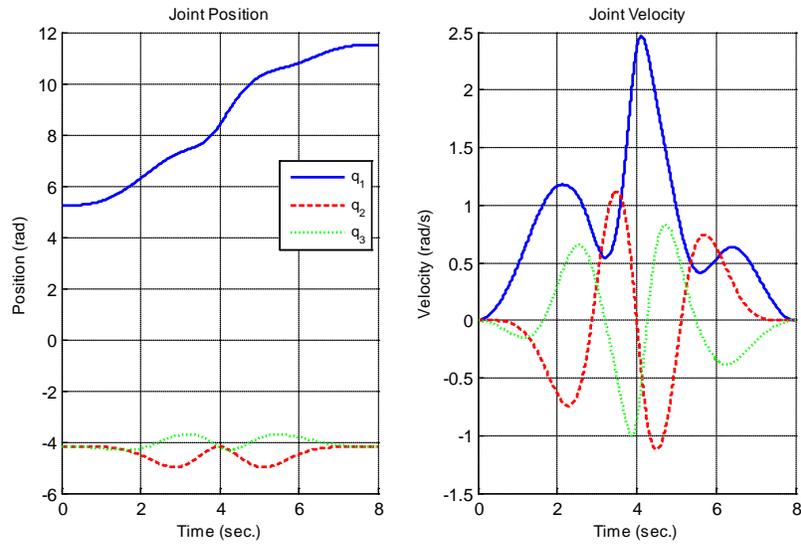


Figure 4.19: Elliptical Contour Joint Positions and Velocities for Serial Manipulator

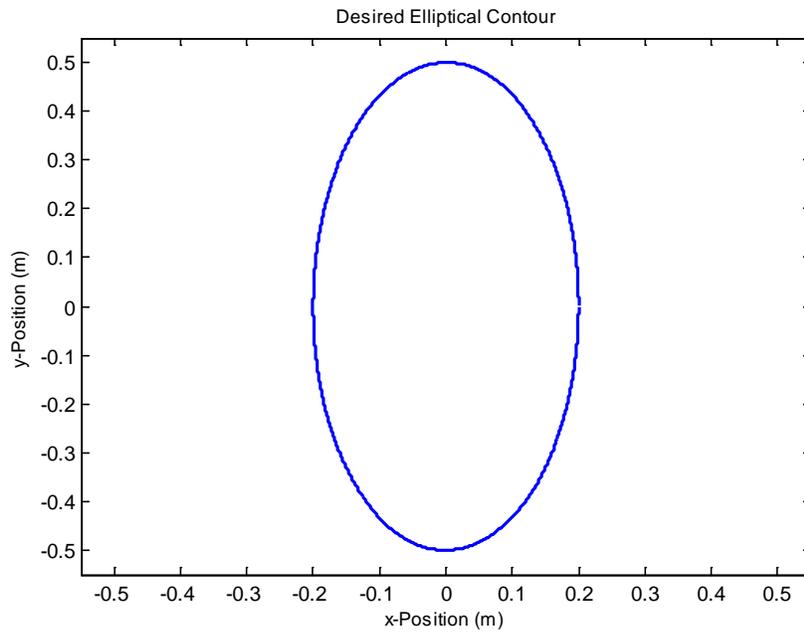


Figure 4.20: Desired Elliptical End-effector Contour for Serial Manipulator

The circular contour information for the parallel manipulator is shown in Table 4.11 and Table 4.12. For the parallel manipulator, two elliptical contours are considered: one that is eccentric in the horizontal x -axis and another that is eccentric in the vertical y -axis. The information for these contours is listed in Table 4.13. The initial and final contour coordinates

for end-effector of the parallel manipulator are given in Table 4.12 and Table 4.14. The task and joint space positions of the parallel manipulator for nonlinear are shown in Figure 4.21 to Figure 4.28.

Table 4.11: Circular Contour Parameters for Parallel Manipulator

x_0 (m)	y_0 (m)	R (m)	Duration (sec)
0.16	0.20	0.06	10

Table 4.12: Circular Contour Initial and Final Positions for Parallel Manipulator

	Initial Value	Final Value
x_{ee} (m)	0.22	0.22
y_{ee} (m)	0.20	0.20

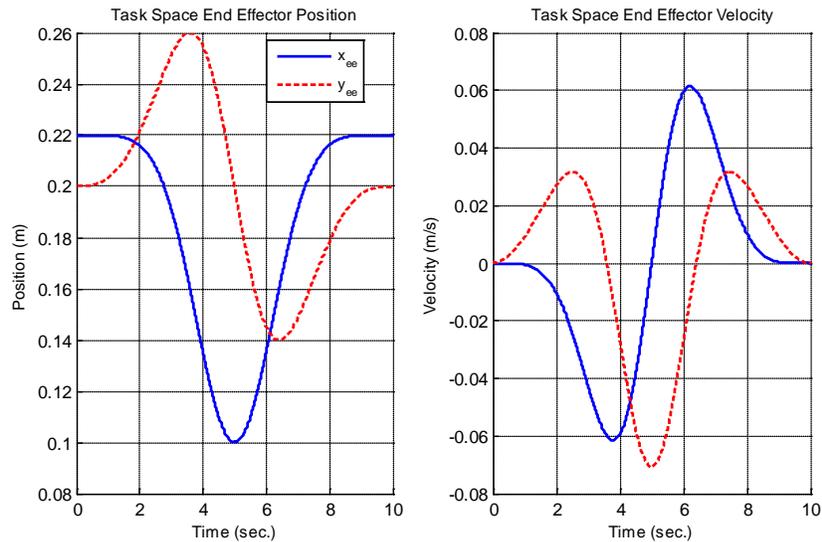


Figure 4.21: Circular Contour Task Space Positions and Velocities for Parallel Manipulator

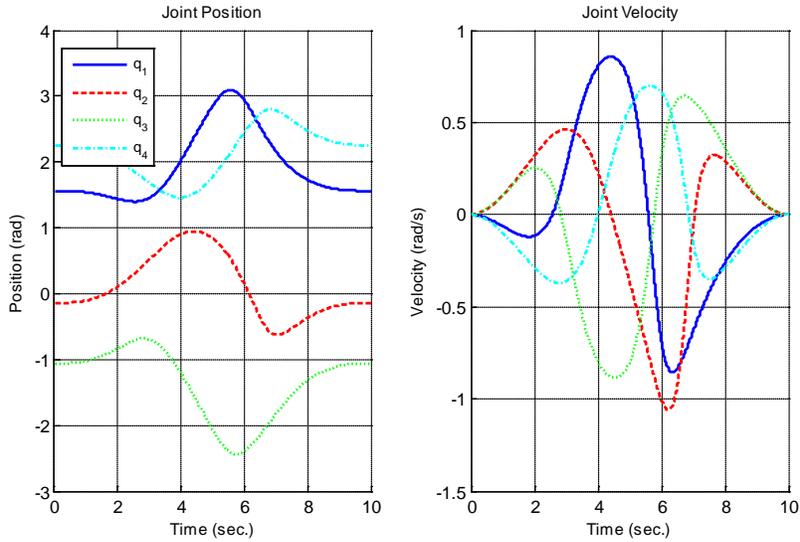


Figure 4.22: Circular Contour Joint Positions and Velocities for Parallel Manipulator

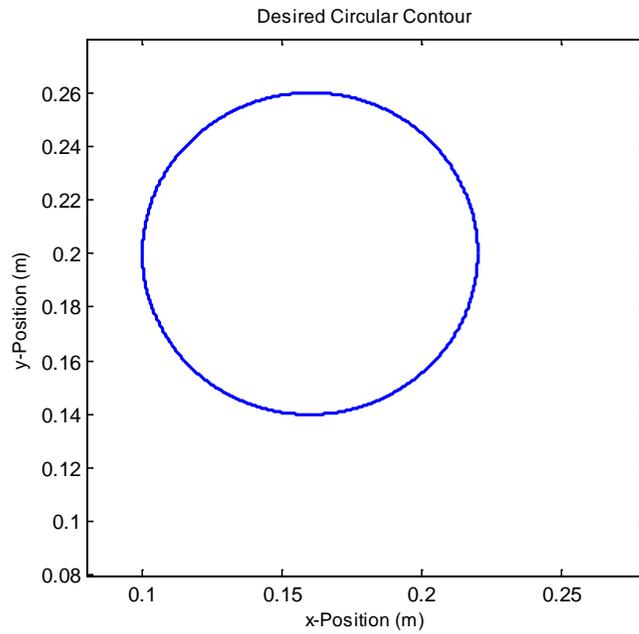


Figure 4.23: Desired Circular End-effector Contour for Parallel Manipulator

Table 4.13: Elliptical Contour Parameters for Parallel Manipulator

	x_0 (m)	y_0 (m)	R_x (m)	R_y (m)	Duration (sec)
Horizontal Ellipse	0.16	0.20	0.06	0.01	6
Vertical Ellipse	0.16	0.20	0.02	0.06	4

Table 4.14: Elliptical Contour Initial and Final Positions for Parallel Manipulator

	Initial Value	Final Value
Horizontal Ellipse		
$x_{ee}(m)$	0.22	0.22
$y_{ee}(m)$	0.20	0.20
Vertical Ellipse		
$x_{ee}(m)$	0.18	0.18
$y_{ee}(m)$	0.20	0.20

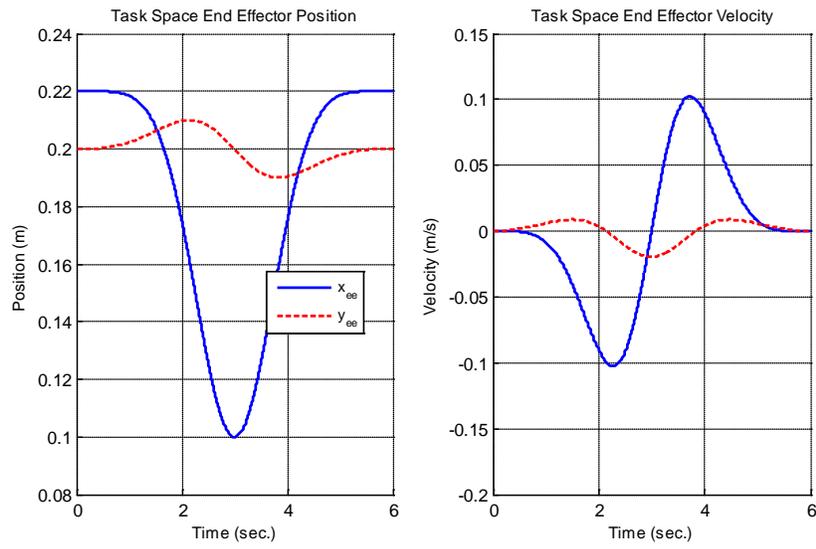


Figure 4.24: Horizontal Elliptical Contour Task Space Positions and Velocities for Parallel Manipulator

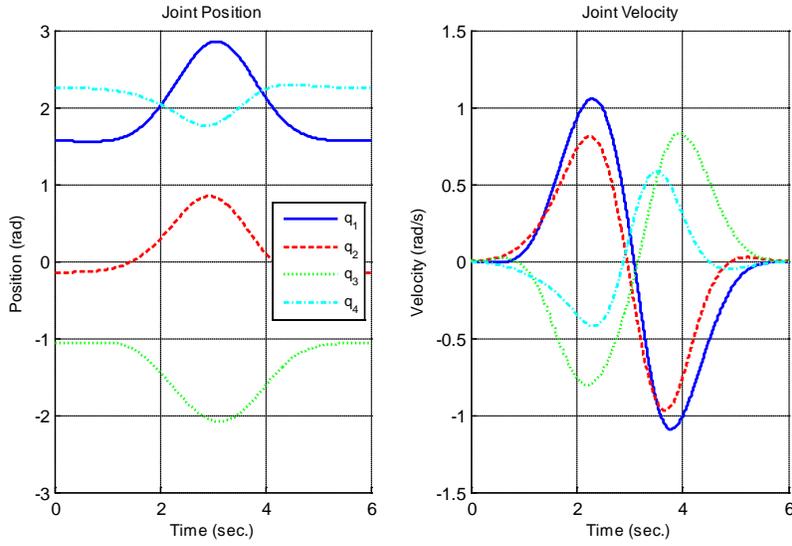


Figure 4.25: Horizontal Elliptical Contour Joint Positions and Velocities for Parallel Manipulator

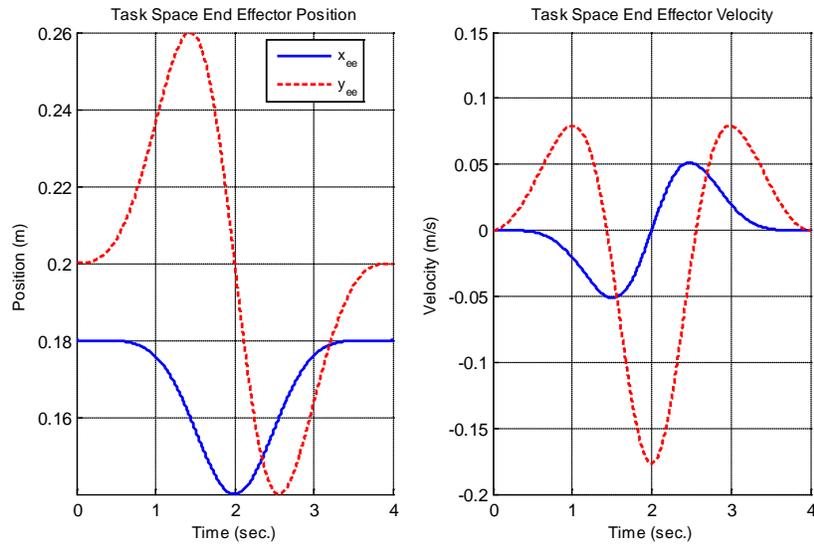


Figure 4.26: Vertical Elliptical Contour Task Space Positions and Velocities for Parallel Manipulator

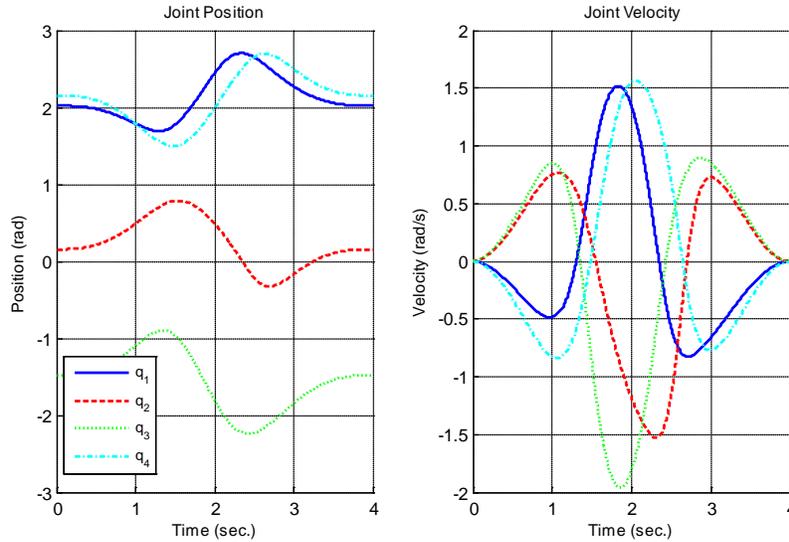


Figure 4.27: Vertical Elliptical Contour Joint Positions and Velocities for Parallel Manipulator

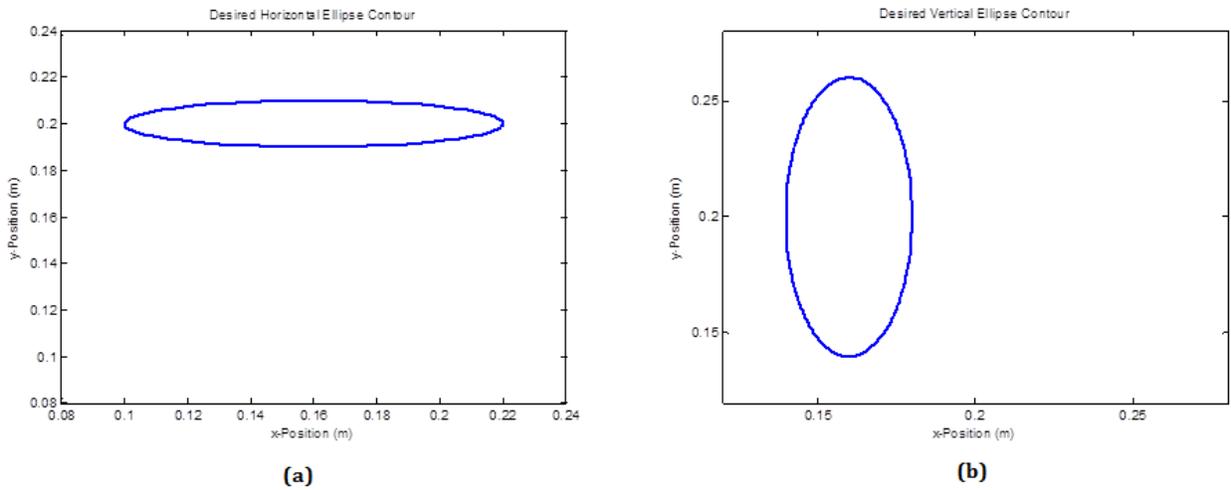


Figure 4.28: Desired Parallel Manipulator End-effector Contours for (a) Horizontal Elliptical (b) Vertical Elliptical

4.2.3 Control Parameter Selection

In order to develop an algorithm for the selection of control parameters, the effect of each parameter must first be determined. This was done by varying each parameter one at a time while maintaining the others at a constant. Simulations were performed for the serial manipulator using a both linear (zigzag) contour and a nonlinear (circular) contour in order to establish the effect of each parameter. The mean contour error of the trajectory for each simulation was used

as an index to measure the relative performance of the manipulator for each parameter variation. For the simulations, each axis uses equal values to one another for any given control parameter, which means each control parameter is a constant diagonal matrix. The baseline control parameters for this simulation are:

$$K_p = \begin{bmatrix} 5000 & 0 & 0 \\ 0 & 5000 & 0 \\ 0 & 0 & 5000 \end{bmatrix} \quad K_d = \begin{bmatrix} 1000 & 0 & 0 \\ 0 & 1000 & 0 \\ 0 & 0 & 1000 \end{bmatrix}$$

$$K_s = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix} \quad \lambda = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

A boundary layer of $\phi = 0.5$ is used for each axis. Table 4.15 shows the ranges of values that were used for each control parameter for the simulations. The increment of each parameter for the following calculation depends on the current value of the parameter. For gains between 0 and 10 the increment is 1, between 10 and 100 the increment is 10, between 100 and 1000 the increment is 100, and between 1000 and 10000 the increment is 1000.

Table 4.15: Control Parameter Variation Ranges

Control Parameter	Minimum Value	Maximum Value
K_p	10	10000
K_d	0	10000
K_s	1	200
λ	0	200

The effects of the variation of the proportional gain (K_p) are shown in Figure 4.29 and Figure 4.30. It is evident that an increase in the proportional gain results in a decrease in the magnitude of joint and end-effector tracking error (e), and a decrease in magnitude of contour error (ε). As K_p becomes large, the decrease in mean contour error and contour error boundaries stabilize. In the case of the serial manipulator for a linear contour, the mean contour error falls

below 10^{-4} [m] at proportional gain values above $K_p \approx 6000$. For the nonlinear contour, although the mean contour error stabilizes quickly, the error boundaries continue to decrease at values over $K_p \approx 8000$.

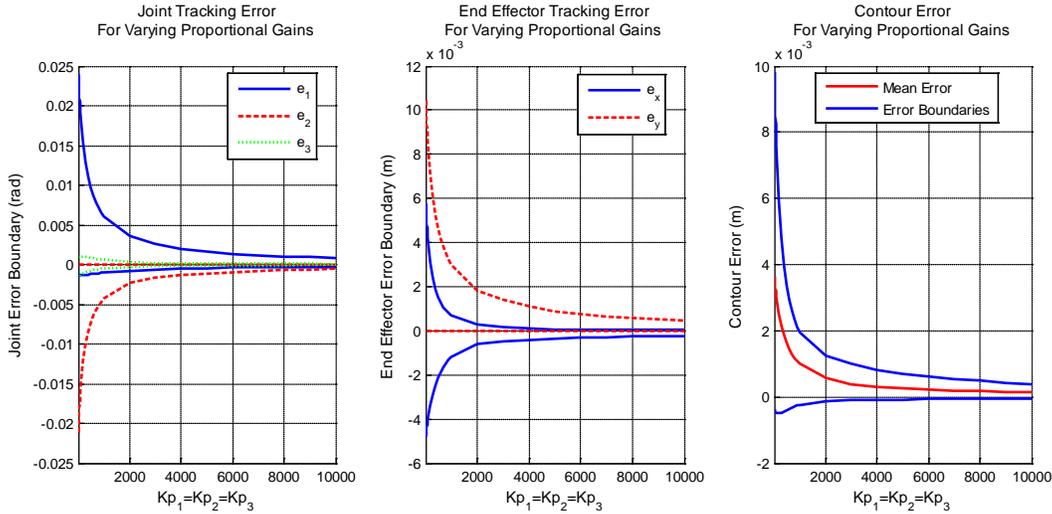


Figure 4.29: Errors for Varying Proportional Gains for a Linear Contour

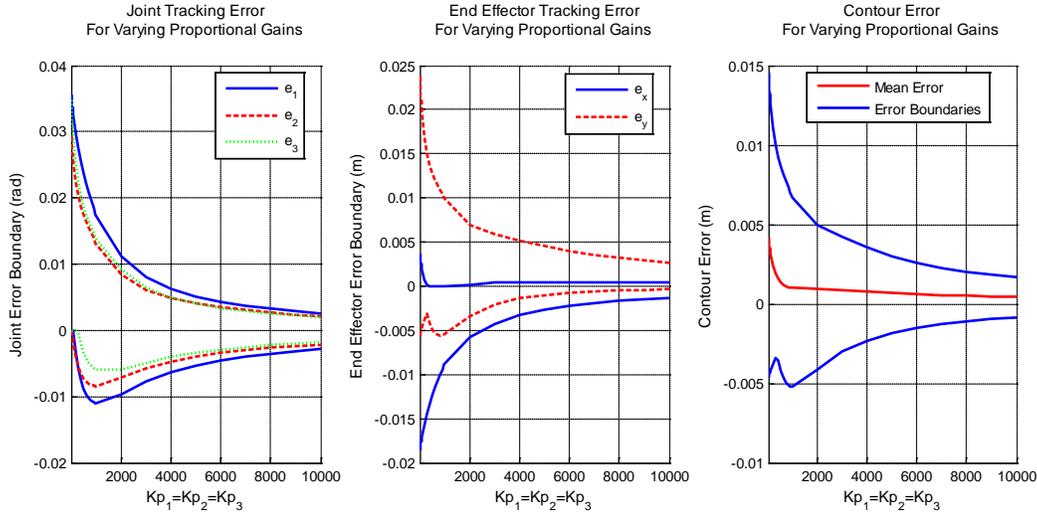


Figure 4.30: Errors for Varying Proportional Gains for a Nonlinear Contour

Figure 4.31 and Figure 4.32 show that an increase in the derivative gain (K_d) also results in a decrease in tracking errors. Although there is a large decrease in the magnitude of the contour error boundaries upon introducing derivative gain, the mean contour tracking

performance is only slightly improved for values larger than $K_d \approx 2000$. It shows that relatively small derivative gains are sufficient to achieve good contour tracking performance.

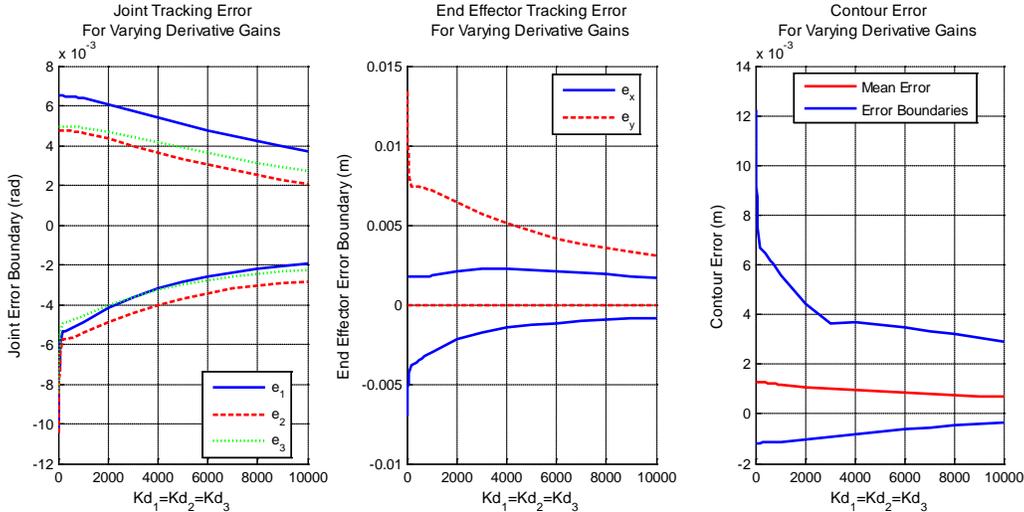


Figure 4.31: Errors for Varying Derivative Gains for a Linear Contour

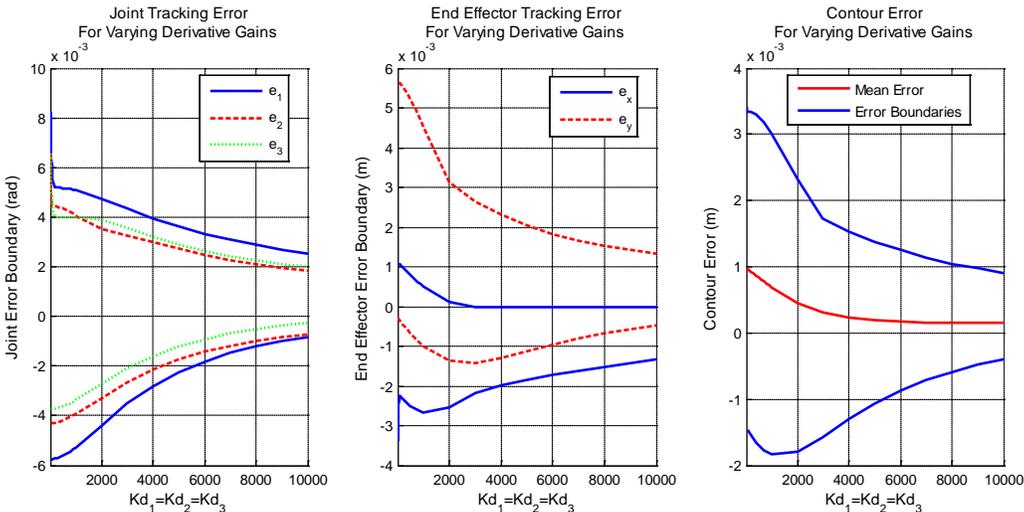


Figure 4.32: Errors for Varying Derivative Gains for a Nonlinear Contour

The effect of varying the SMC gain (K_s) are shown in Figure 4.33 and Figure 4.34. It can be seen that the tracking and contour errors decrease with increasing SMC gain. While the error boundaries show a reasonable decrease with larger values of K_s , the mean contour error does not change significantly and falls below 10^{-4} [m] at values of SMC gain above $K_s \approx 50$ for these control parameter values.

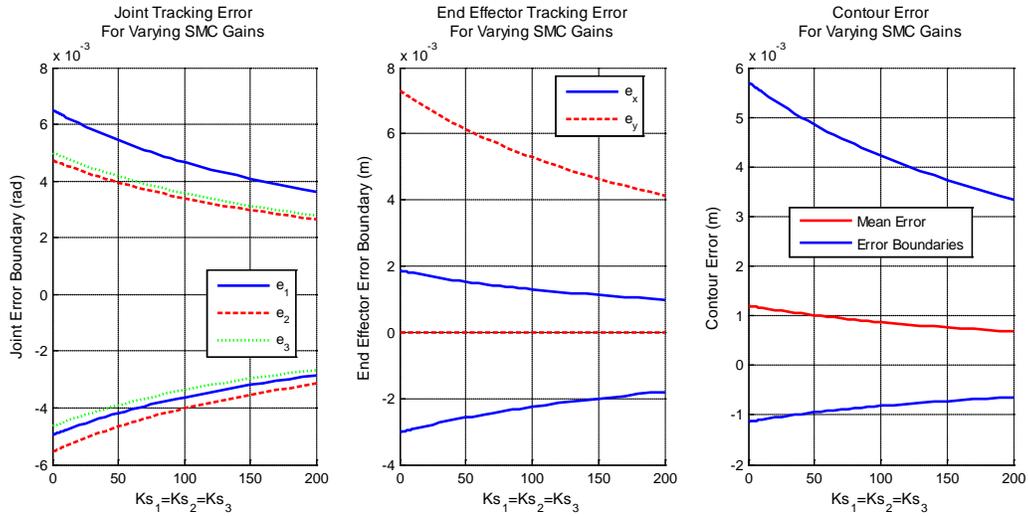


Figure 4.33: Errors for Varying SMC Gains for a Linear Contour

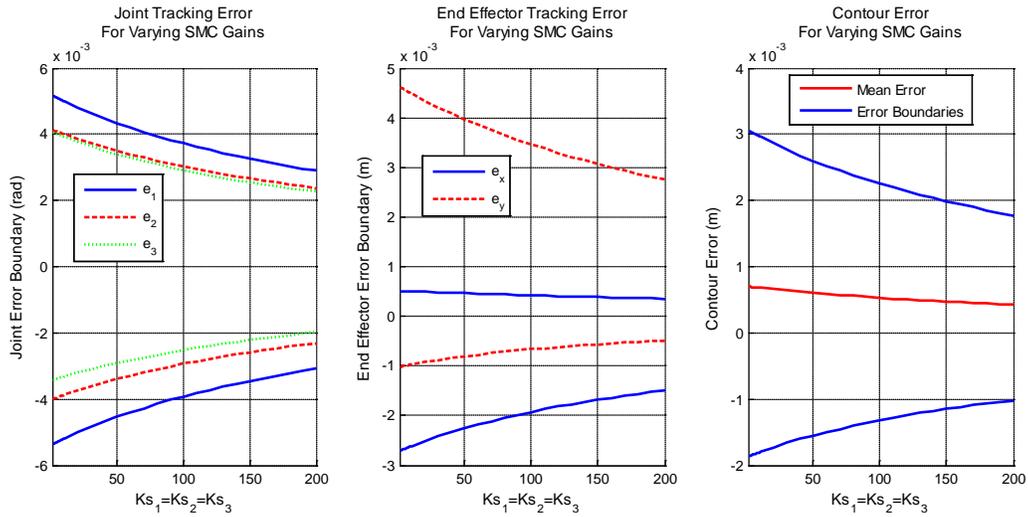


Figure 4.34: Errors for Varying SMC Gains for a Nonlinear Contour

Finally, the effects of varying the bandwidths (λ) of the controllers are shown in Figure 4.35 and Figure 4.36. Although there is a decrease in mean contour error resulting from an increase in bandwidth, the effect is relatively small. The error boundaries, however, are shown to decrease in magnitude until a certain point at which the change in error stabilizes.

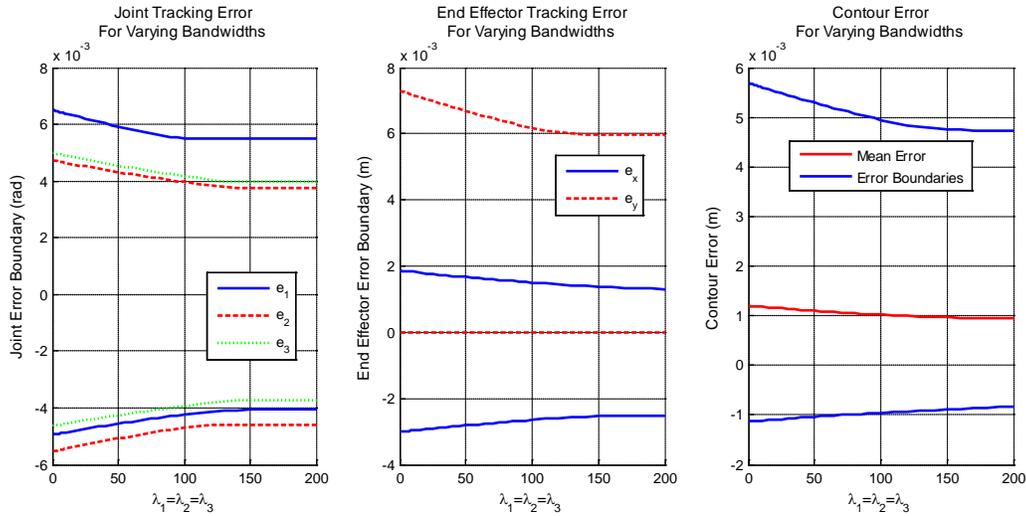


Figure 4.35: Errors for Varying Bandwidths for a Linear Contour

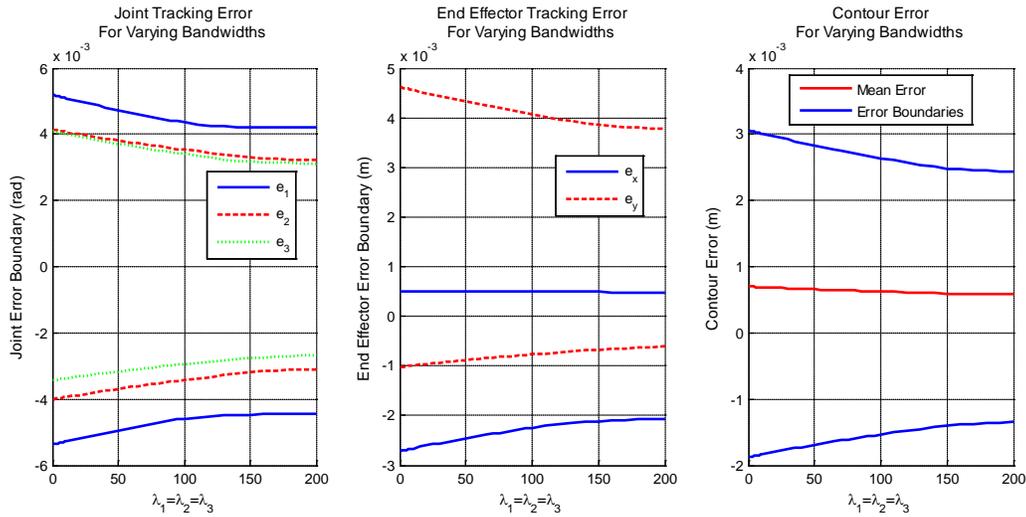


Figure 4.36: Errors for Varying Bandwidths for a Nonlinear Contour

Although equal gains are used for all the active joints, in practice the gains for each joint may differ from one another and may be tuned for individual performance. In this case, it is expected that the gains for axis 2 and axis 3 of the serial manipulator will be smaller than those of axis 1 since the added masses and inertias for the latter axes will require larger control gains for axis 1, the base joint.

It should be noted that the behaviour of the system with varying proportional and derivative gains is similar to that of a PD-controlled system. A difference in interpretation,

however, should be noted in that K_p and K_d serve to replace the equivalent dynamics of a pure SMC system. In this regard, instead of the controller being tuned to produce zero tracking error directly, the proportional and derivative gains can be seen as “pushing” the system onto the sliding surface such that $\dot{s} = 0$. Also, from the definition of the sliding surface Eq. (3.6), the bandwidth can be interpreted as a scaling factor for the joint position and velocity errors. As such, λ should initially be selected such that the position and velocity errors are of roughly the same magnitude.

Based on the observed behaviour of the hybrid control law, the following algorithm can be used for simple tuning of the hybrid PD-SMC law:

1. Set $K_s = 0$, and select K_p and K_d by trial and error or employing an algorithm for PD-type controllers.
2. Select λ such that the position errors are scaled to roughly the same magnitudes as the velocity errors.
3. Increase K_s until desired contour performance is achieved.
4. Select a boundary layer thickness, ϕ , to smooth the control torque input, as desired.
5. Fine tune control parameters, as needed.

4.3 SERIAL MANIPULATOR SIMULATION RESULTS

The control parameters used for the serial manipulator in the following simulations are listed in Table 4.16. Boundary layer thicknesses were applied to smooth the control torque inputs for the SMC and the hybrid PD-SMC controllers, as desired, and are also listed in the table.

Table 4.16: Control Parameters for Serial Manipulator Simulations

Controller Type	Linear Contours	Circular Contour	Elliptical Contour
PD	$K_p = \text{diag}\{7700,7500,2100\}$	$K_p = \text{diag}\{9200,8300,2200\}$	$K_p = \text{diag}\{8200,8200,1800\}$
	$K_d = \text{diag}\{2000,1200,1200\}$	$K_d = \text{diag}\{2400,1600,1400\}$	$K_d = \text{diag}\{1800,1100,1100\}$
SMC	$K_s = \text{diag}\{65,30,20\}$	$K_s = \text{diag}\{55,35,25\}$	$K_s = \text{diag}\{45,20,18\}$
	$\lambda = \text{diag}\{25,15,10\}$	$\lambda = \text{diag}\{22,15,10\}$	$\lambda = \text{diag}\{25,20,10\}$
	$\phi = \text{diag}\{0.25,0.3,0.25\}$	$\phi = \text{diag}\{0.25,0.3,0.25\}$	$\phi = \text{diag}\{0.25,0.3,0.25\}$
PD-SMC	$K_p = \text{diag}\{7700,7500,2100\}$	$K_p = \text{diag}\{9200,8300,2200\}$	$K_p = \text{diag}\{8200,8200,1800\}$
	$K_d = \text{diag}\{2000,1200,1200\}$	$K_d = \text{diag}\{2400,1600,1400\}$	$K_d = \text{diag}\{1800,1100,1100\}$
	$K_s = \text{diag}\{65,30,20\}$	$K_s = \text{diag}\{55,35,25\}$	$K_s = \text{diag}\{45,20,18\}$
	$\lambda = \text{diag}\{25,15,10\}$	$\lambda = \text{diag}\{22,15,10\}$	$\lambda = \text{diag}\{25,20,10\}$
	$\phi = \text{diag}\{0.25,0.3,0.25\}$	$\phi = \text{diag}\{0.25,0.3,0.25\}$	$\phi = \text{diag}\{0.25,0.3,0.25\}$

For the standard SMC simulations, the uncertainties of the dynamic models are applied by using the following estimations of parameter matrices:

$$\widehat{M}(q) = 0.9\bar{M}(q) \quad \widehat{C}(q, \dot{q}) = 0.9C(q, \dot{q}) \quad \widehat{G}(q) = 1.1G(q)$$

4.3.1 Zigzag Contour

Using the controller gains listed in Table 4.16, good axial tracking performance was achieved for all three control schemes: PD, SMC, and PD-SMC in the simulation of the zigzag contour. It should be noted that the means and standard deviations (S.D.) recorded for each of the simulations were calculated from the absolute values of the errors. From the results displayed in Table 4.17, it can be seen that both the SMC and hybrid PD-SMC controller achieve better axial tracking performances than the standard PD controller. Although the PD-SMC controller achieved 44.1% (axis 1), 16.6% (axis 2), and 24.8% (axis 3) smaller mean axial tracking errors than the PD controller, the SMC controller was able to outperform the PD-SMC controller in

both axis 1 and axis 3. The SMC controller, however, achieved an 18% lower mean axial tracking error in axis 2. The joint space tracking errors can also be seen in Figure 4.37.

Table 4.17: Mean and Standard Deviation of Axial Tracking Errors for Zigzag Motion of Serial Manipulator

	Axis 1		Axis 2		Axis 3	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.0005401	0.0002137	0.0005415	0.0001434	0.0002236	$8.349(10^{-5})$
SMC	0.0001287	$5.159(10^{-5})$	0.0005503	0.0001393	0.0001420	$7.418(10^{-5})$
PD-SMC	0.0003020	0.0001169	0.0004515	0.0001204	0.0001682	$6.053(10^{-5})$

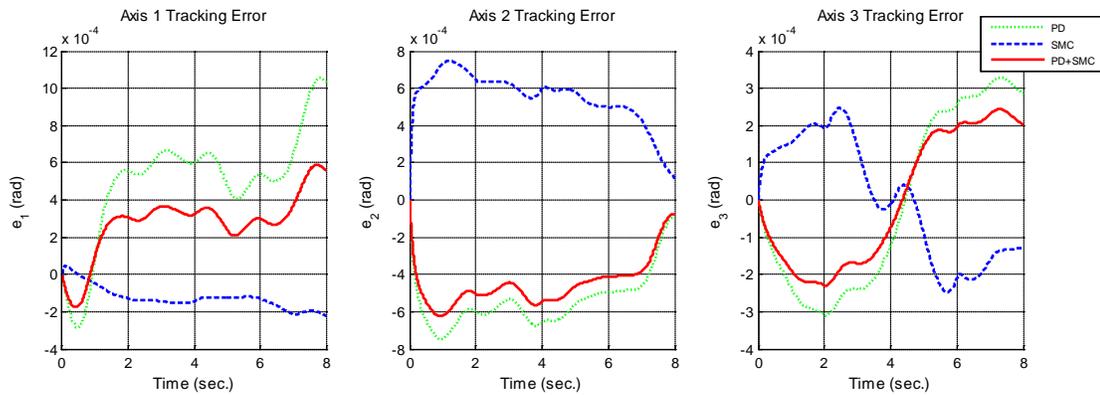


Figure 4.37: Axial Tracking Error for Zigzag Motion of Serial Manipulator

Also of interest is the tracking error of the end-effector in the task space. The means and standard deviations of these errors for the zigzag contour are listed in Table 4.18. From the displayed results it can be seen that the PD-SMC controller achieves better task space tracking performance of the end-effector than both the standard PD and SMC controllers. The hybrid PD-SMC controller yielded 8.5% better total tracking performance than the SMC controller and 24.2% better performance than the PD controller. The end-effector performance in the task space as well as the end-effector pose tracking error is shown in Figure 4.38.

Table 4.18: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Zigzag Motion of Serial Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	0.0001230	$5.215(10^{-5})$	0.0003978	0.0001208	0.0003978	0.0001180
SMC	$8.503(10^{-5})$	$6.756(10^{-5})$	0.0003138	0.0001534	0.0003295	0.0001588
PD-SMC	$7.003(10^{-5})$	$7.001(10^{-5})$	0.0002896	$9.47(10^{-5})$	0.0003015	$9.952(10^{-5})$

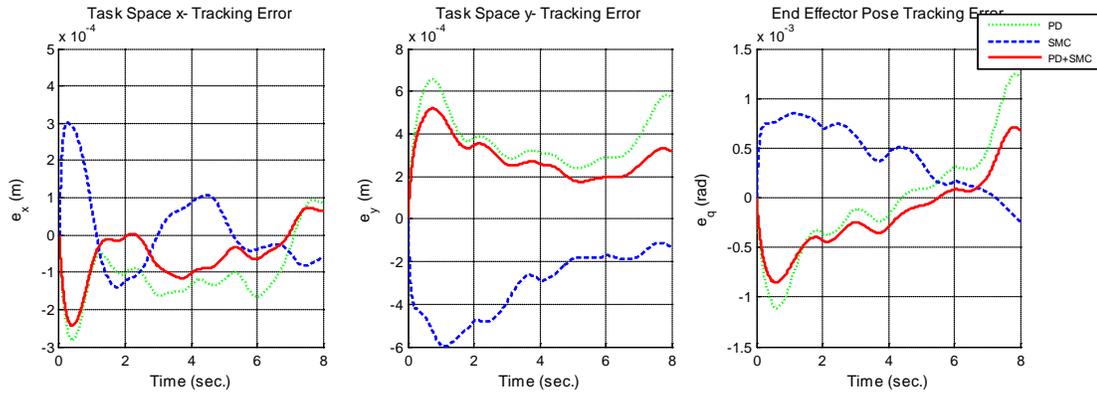


Figure 4.38: End-Effector Task Space Tracking Error for Zigzag Motion of the Serial Manipulator

The contour errors of each of the three controllers are displayed in Figure 4.39. Table 4.19 also lists the contour tracking performance of the controllers. From the results shown in Table 4.19, it can be seen that the hybrid PD-SMC controller achieves better contour tracking performance than both the PD and SMC controllers. The PD-SMC controller produced 19.8% smaller contour errors than the PD controller and 13.2% smaller contour errors than the SMC controller.

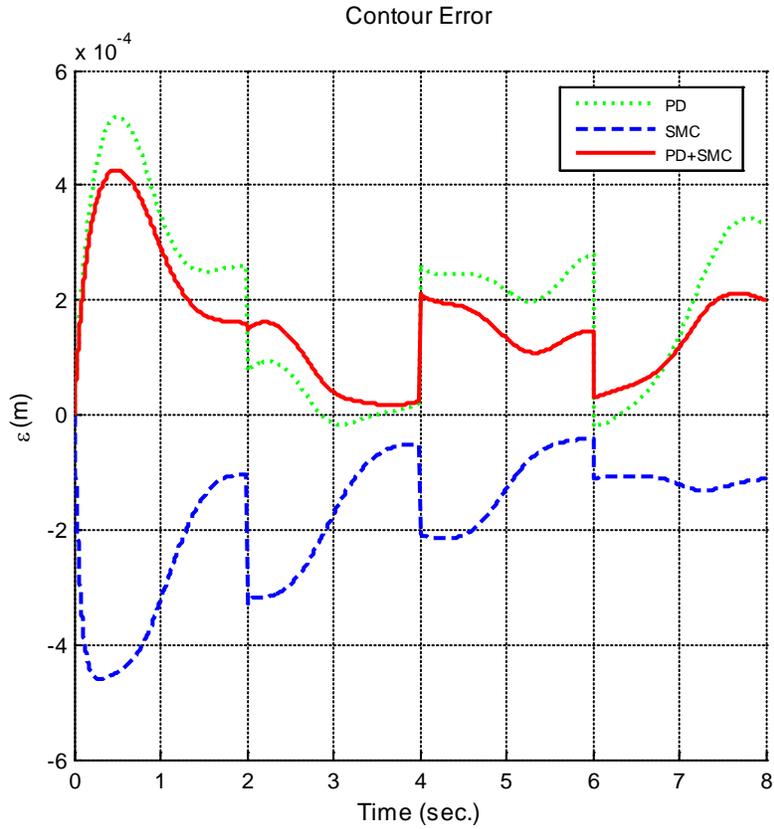


Figure 4.39: Contour Error for Zigzag Motion of the Serial Manipulator

Table 4.19: Magnitude of Contour Tracking Performance for Zigzag Motion of Serial Manipulator

	Mean (m)	S.D. (m)	$\%(\epsilon \leq 2(10^{-4}) [m])$
PD	0.0001928	0.0001427	42.71%
SMC	0.0001781	0.0001142	65.28%
PD-SMC	0.0001546	0.0001013	75.96%

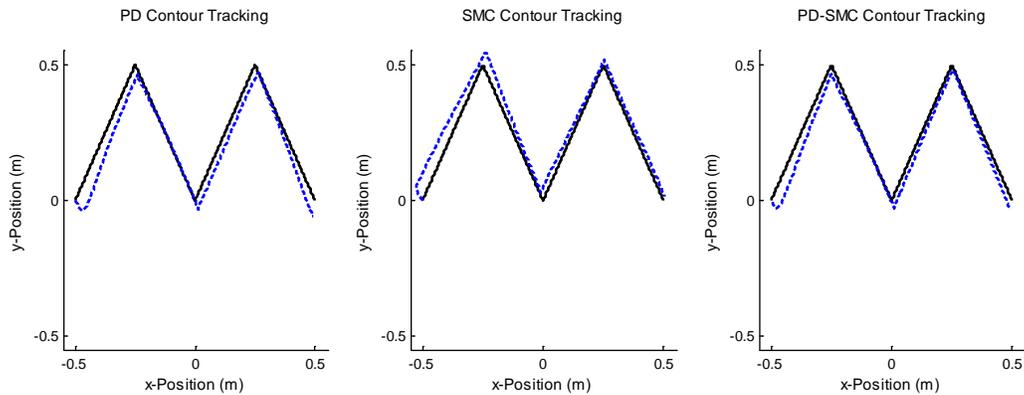


Figure 4.40: Zigzag Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator¹

To further illustrate the relative performance of the control laws, the amount of the motion that falls within a boundary of error of $2(10^{-4})$ [m] was also recorded. That is, the percent of the motion that the magnitude of contour error is smaller than $2(10^{-4})$ [m] is also listed in Table 4.19. While only 42.7% and 65.3% of the points traced along the zigzag contour fall within this boundary of error for the PD and SMC controllers, respectively, 76.0% of the contour errors for the PD-SMC controller are shown to be within this bound. Plots of the desired and actual contours are shown in Figure 4.40.

The torque results listed in Table 4.20 show that for axis 1 and axis 3, the SMC controller produced the smallest maximum values of input torque, while the PD-SMC controller produced the largest maximum torques. For the second axis, the maximum input torques remained approximately equal for all three controllers. The torques produced by each of the controllers for each axis is shown in Figure 4.41.

¹ Plots include an error magnification factor of 100

Table 4.20: Maximum Input Torques for Zigzag Motion of Serial Manipulator

	τ_1 (Nm)	τ_2 (Nm)	τ_3 (Nm)
PD	9.075	5.558	0.8275
SMC	8.875	5.762	0.7708
PD-SMC	10.44	5.725	0.8796

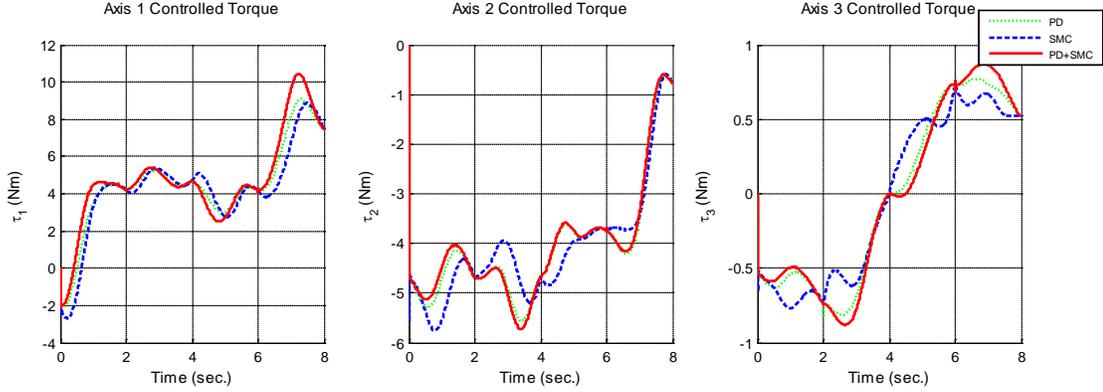


Figure 4.41: Input Torques for Zigzag Motion of Serial Manipulator

4.3.2 Diamond Contour

Similar to the zigzag contour, the results of the axial tracking performance for the diamond contour were good for all three simulated controllers. Once again the PD-SMC achieved better axial tracking results than the standard PD controller, as shown in Table 4.21. The PD-SMC yielded lower mean axial tracking errors than the PD controller by 43.7% (axis 1), 16.4% (axis 2), and 23.6% (axis 3). The SMC controller produced the best axial tracking results in axis 1 and axis 3 again, but was outperformed by the PD-SMC controller by 17.4% in axis 2. The standard deviations of the mean errors observed the same trend as the mean error magnitudes. Figure 4.42 shows the tracking errors in the joint space for the diamond contour.

Table 4.21: Mean and Standard Deviation of Axial Tracking Errors for Diamond Motion of Serial Manipulator

	Axis 1		Axis 2		Axis 3	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.0005707	0.0002213	0.0003579	0.0002009	0.0001857	0.0001194
SMC	0.0001453	5.528(10 ⁻⁵)	0.0003260	0.0002039	0.0001231	7.248(10 ⁻⁵)
PD-SMC	0.0003211	0.0001231	0.0002991	0.0001680	0.0001419	9.346(10 ⁻⁵)

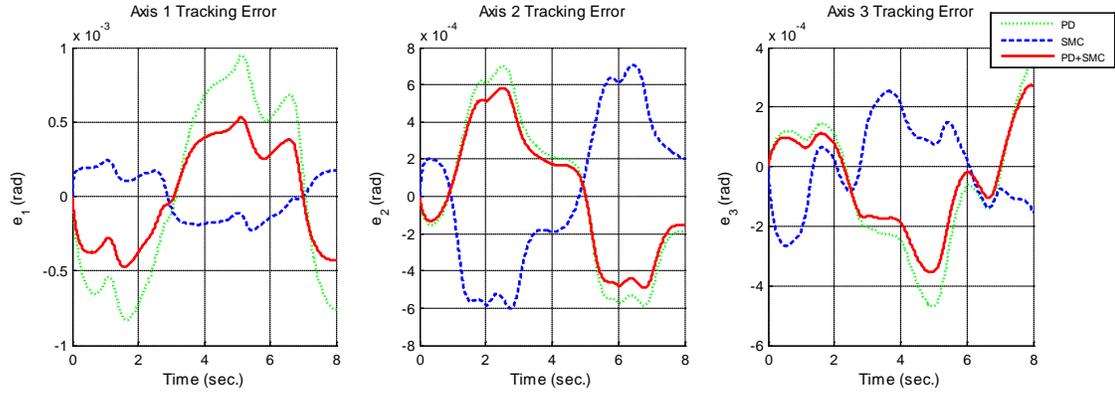


Figure 4.42: Axial Tracking Error for Diamond Motion of Serial Manipulator

From the results listed in Table 4.22, it is seen that the PD-SMC controller achieved the best end-effector task space tracking performance for the diamond contour as well. The hybrid controller produced 28.3% smaller total tracking error magnitudes than the PD controller and 17.6% smaller tracking error magnitudes than the SMC controller. The PD-SMC also produced a smaller standard deviation for the total tracking error magnitude, outperforming the PD controller by 22.1% and the SMC controller by 46.6% in this field. The end-effector tracking errors are shown in Figure 4.43.

Table 4.22: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Diamond Motion of Serial Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	0.0001196	5.461(10 ⁻⁵)	0.0001750	7.720(10 ⁻⁵)	0.0002197	7.483(10 ⁻⁵)
SMC	0.0001164	6.294(10 ⁻⁵)	0.0001458	9.889(10 ⁻⁵)	0.0001913	0.0001092
PD-SMC	6.199(10 ⁻⁵)	5.625(10 ⁻⁵)	0.0001376	6.768(10 ⁻⁵)	0.0001576	5.833(10 ⁻⁵)

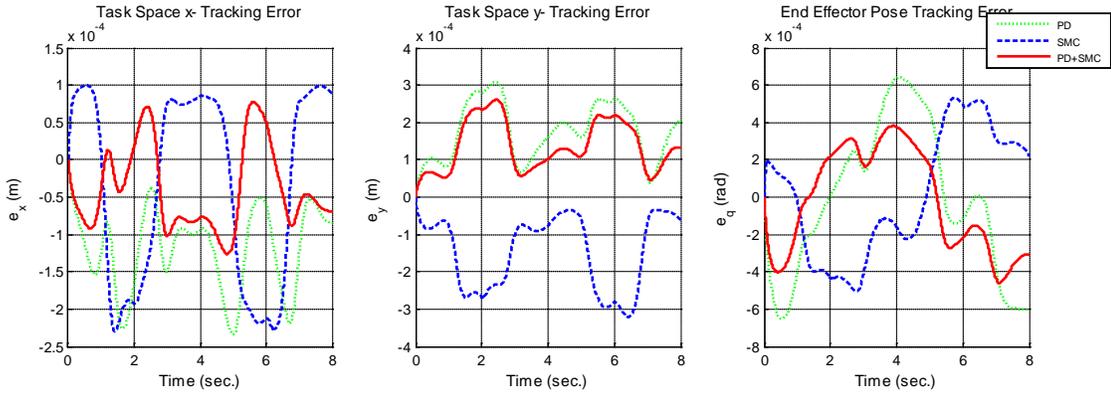


Figure 4.43: End-Effector Task Space Tracking Error for Diamond Motion of the Serial Manipulator

In terms of contour tracking performances, the PD-SMC controller once again outperforms both of the other control schemes. The mean contour error magnitude for the PD-SMC controller was more than 40% lower than those of the PD and SMC controllers. The standard deviation of these errors for the PD-SMC controller was also lower than the other two controllers by 38.2% and 53.4% for the PD and SMC controllers, respectively. These results indicate that the percent of the motion that the contour error magnitude is smaller than $2(10^{-4}) [m]$ along the diamond contour for the PD-SMC controller is 100%. The corresponding percentages for the PD and SMC controller are 93% and 81%, respectively. The contour error for this simulation can be seen in Figure 4.44 while the plots of the desired and actual contours are shown in Figure 4.45.

Table 4.23: Magnitude of Contour Tracking Performance for Diamond Motion of Serial Manipulator

	Mean (<i>m</i>)	S.D. (<i>m</i>)	%(ϵ $\leq 2(10^{-4}) [m]$)
PD	0.0001163	$5.619(10^{-5})$	92.82%
SMC	0.0001177	$7.469(10^{-5})$	80.79%
PD-SMC	$6.518(10^{-5})$	$3.472(10^{-5})$	100%

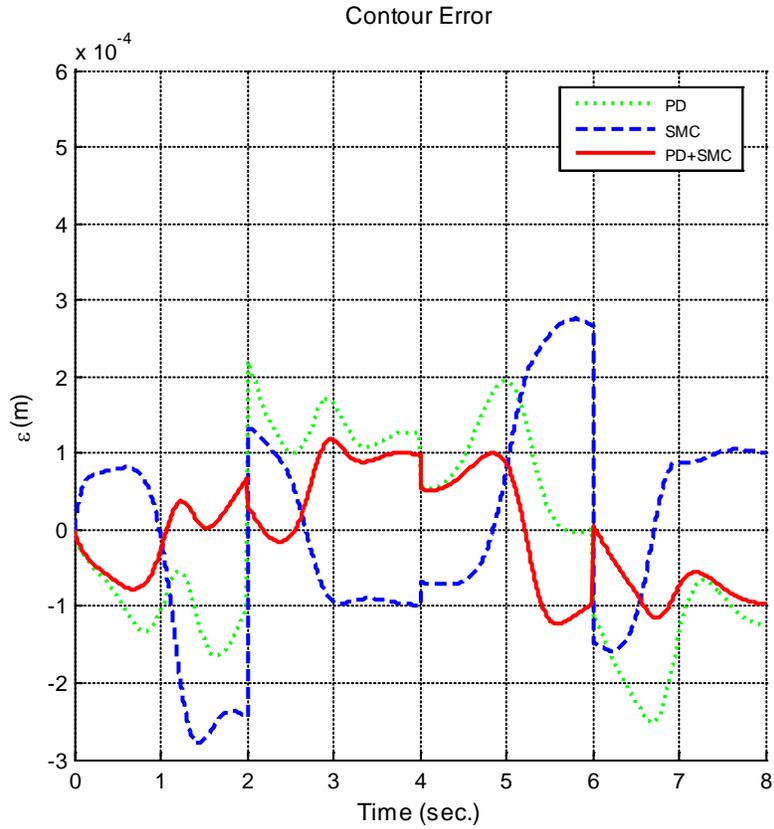


Figure 4.44: Contour Error for Diamond Motion of the Serial Manipulator

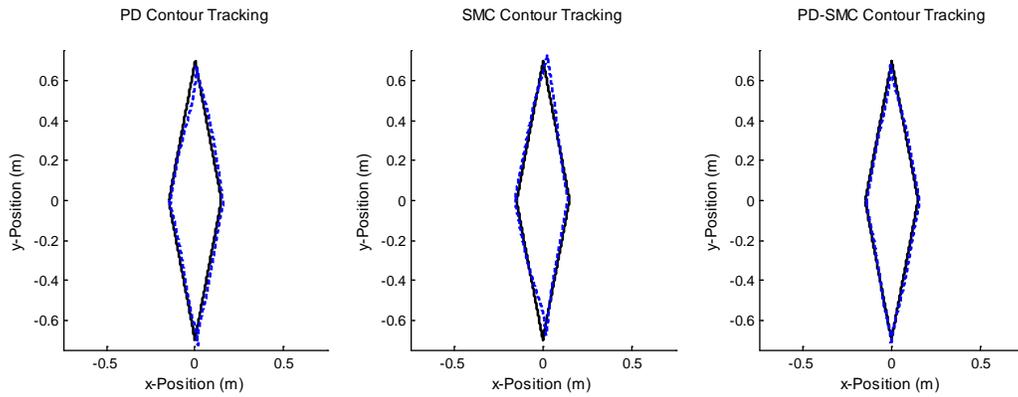


Figure 4.45: Diamond Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator²

From the results shown in Table 4.24 and Figure 4.46, it is seen that all three controllers exhibit similar torque requirements for the diamond contour. The largest deviation in input

² Plots include an error magnification factor of 100

torques is for axis 3, where both the PD and PD-SMC controllers produce about 10% less maximum torque than the SMC controller.

Table 4.24: Maximum Input Torques for Diamond Motion of Serial Manipulator

	τ_1 (Nm)	τ_2 (Nm)	τ_3 (Nm)
PD	7.797	5.310	1.040
SMC	7.965	5.346	1.164
PD-SMC	7.679	5.304	1.009

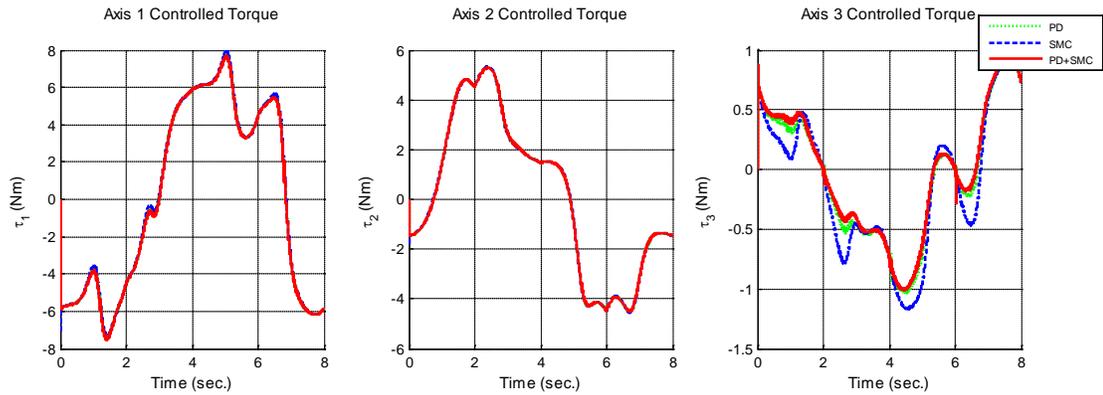


Figure 4.46: Input Torques for Diamond Motion of Serial Manipulator

4.3.3 Circular Contour

The simulations for the circular contour produced good axial tracking results for each controller, as indicated in Table 4.25. The same trend in relative axial tracking performance is once again present as the SMC controller yielded the best results for both axis 1 and axis 3. While the PD-SMC controller produced an estimated 30% smaller mean error magnitude and standard deviation of this error than the standard PD controller in these axes, the SMC controller yielded 30% and 20% smaller mean error magnitudes than the PD-SMC controller in axis 1 and axis 3, respectively. However, the PD-SMC controller once again had the best performance in axis 2, producing 16.4% and 26.6% smaller mean error magnitudes than the PD and SMC controllers, respectively. The axial tracking errors for the three controllers are displayed in Figure 4.47.

Table 4.25: Mean and Standard Deviation of Axial Tracking Errors for Circular Motion of Serial Manipulator

	Axis 1		Axis 2		Axis 3	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.0005136	0.0002821	0.0003145	0.0001535	0.0002520	6.296(10 ⁻⁵)
SMC	0.0002312	7.856(10 ⁻⁵)	0.0003585	0.0001651	0.0001436	3.267(10 ⁻⁵)
PD-SMC	0.0003485	0.0001932	0.0002630	0.0001273	0.0001806	4.260(10 ⁻⁵)

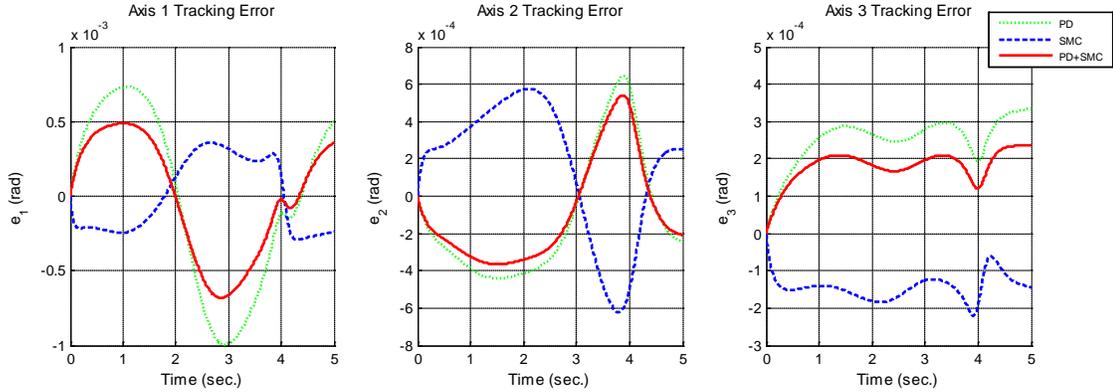


Figure 4.47: Axial Tracking Error for Circular Motion of Serial Manipulator

The results for the magnitude of tracking error in the task space for the simulations are shown in Table 4.26. The PD-SMC controller produced better end-effector tracking results than the PD controller in both the x and y axes by 33.9% and 22.5%, respectively, in terms of the mean error magnitudes. The SMC controller, however, yielded better results than the PD-SMC in these fields by about 2% and 9% in the x and y axes, respectively. In terms of total task space tracking error magnitude, the PD-SMC controller produced a 24.5% smaller mean than the PD, while the SMC produced a 7.9% smaller mean than PD-SMC. The PD-SMC, however, yielded a 6.5% smaller standard deviation of error magnitudes than the SMC controller. The task space tracking errors for the three controllers are displayed in Figure 4.48.

Table 4.26: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Circular Motion of Serial Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	0.0001032	$8.594(10^{-5})$	0.0002655	0.0001053	0.0002970	0.0001068
SMC	$6.684(10^{-5})$	$5.257(10^{-5})$	0.0001882	$8.812(10^{-5})$	0.0002065	$8.798(10^{-5})$
PD-SMC	$6.818(10^{-5})$	$7.672(10^{-5})$	0.0002059	$8.173(10^{-5})$	0.0002242	$8.224(10^{-5})$

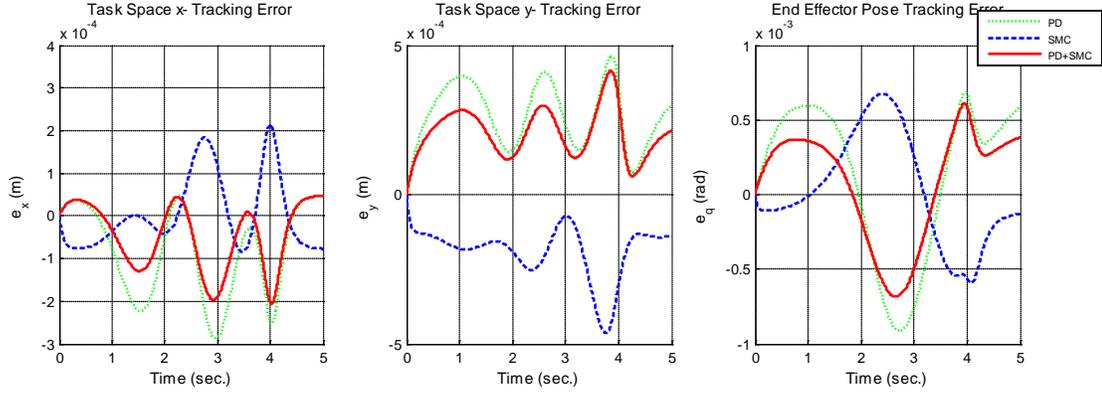


Figure 4.48: End-Effector Task Space Tracking Error for Circular Motion of the Serial Manipulator

It should be noted that for all nonlinear contours, the contour error was estimated by locally approximating the curve as a circle and using Eq. (2.10). The results listed in Table 4.27 indicate that the contour tracking performance of the PD-SMC controller is superior to both the PD and SMC controllers. The mean of the magnitudes of the contour error for the PD-SMC controller is 16.8% smaller than that of the PD controller and 24.3% smaller than that of the SMC controller. The standard deviation of these errors for the PD-SMC controller also yielded the best results of the three controllers. Both the SMC and PD-SMC controllers produced a more than 15% smaller standard deviation of contour error magnitudes than the standard PD controller. Also, for the performance of this circular contour, the contour errors yield about 84.5% of all points smaller than a $2(10^{-4})$ [m] contour error for the PD controller, 85.4% for the SMC controller, and 88.4% for the PD-SMC controller. The contour errors for the

simulations are displayed in Figure 4.49 while the desired and actual plots are displayed in Figure 4.50.

Table 4.27: Magnitude of Contour Tracking Performance for Circular Motion of Serial Manipulator

	Mean (m)	S.D. (m)	$\%(\epsilon \leq 2(10^{-4}) [m])$
PD	0.0001133	$9.136(10^{-5})$	84.45%
SMC	0.0001244	$7.698(10^{-5})$	85.37%
PD-SMC	$9.422(10^{-5})$	$7.545(10^{-5})$	88.36%

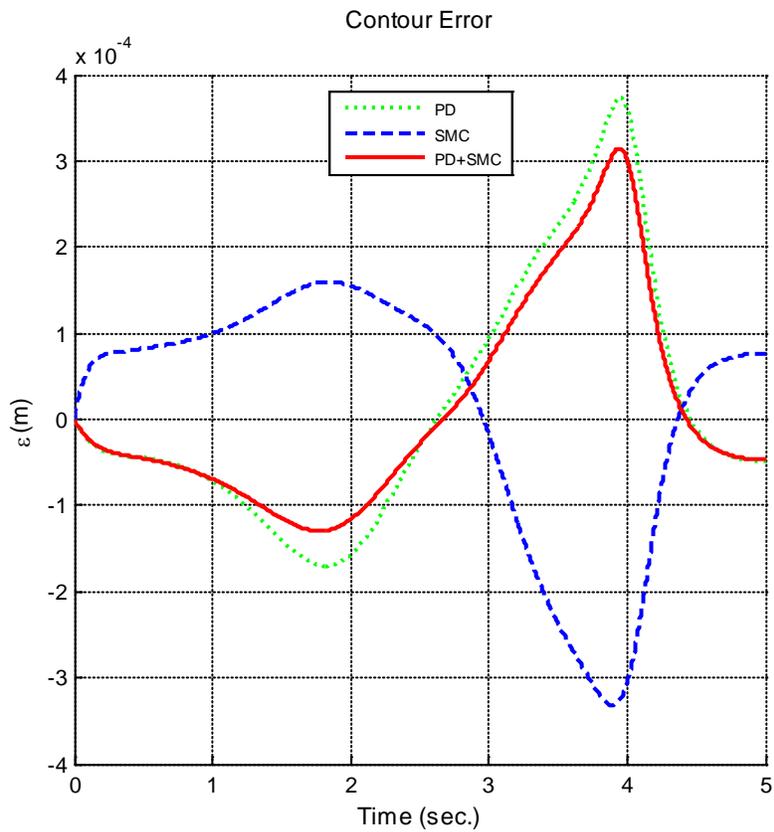


Figure 4.49: Contour Error for Circular Motion of the Serial Manipulator

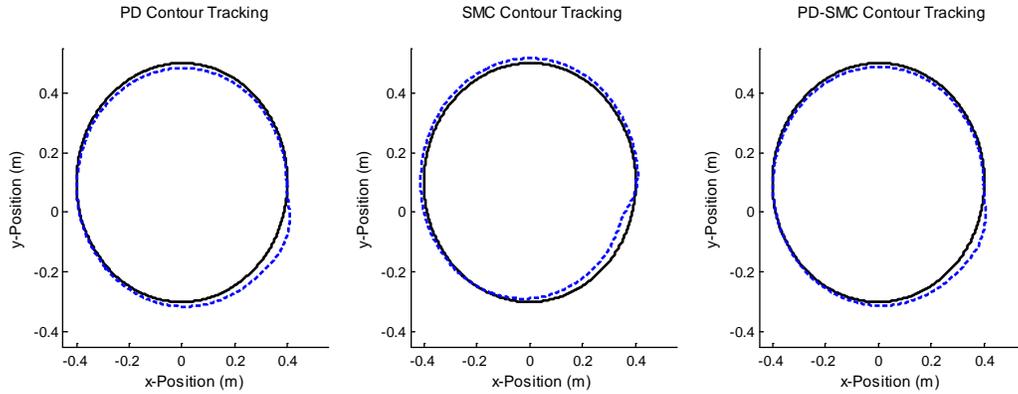


Figure 4.50: Circular Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator³

The maximum torques required for each controller are listed in Table 4.28. The maximum torque required for axis 1 is largest for the PD-SMC controller, but for axis 2 and axis 3, the SMC controller produced the largest maximum torque. For axis 1, the maximum torque required of the SMC controller is 20.6% lower than the maximum torque required of the PD-SMC controller. The PD-SMC controller has a maximum torque requirement that is 23.4% smaller than that of the SMC controller for axis 2 and 42.8% for axis 3. The controlled torques for each of the three controllers is displayed in Figure 4.51.

Table 4.28: Maximum Input Torques for Circular Motion of Serial Manipulator

	τ_1 (Nm)	τ_2 (Nm)	τ_3 (Nm)
PD	11.91	4.979	0.8795
SMC	10.28	5.795	1.690
PD-SMC	12.95	4.440	0.9660

³ Plots include an error magnification factor of 100

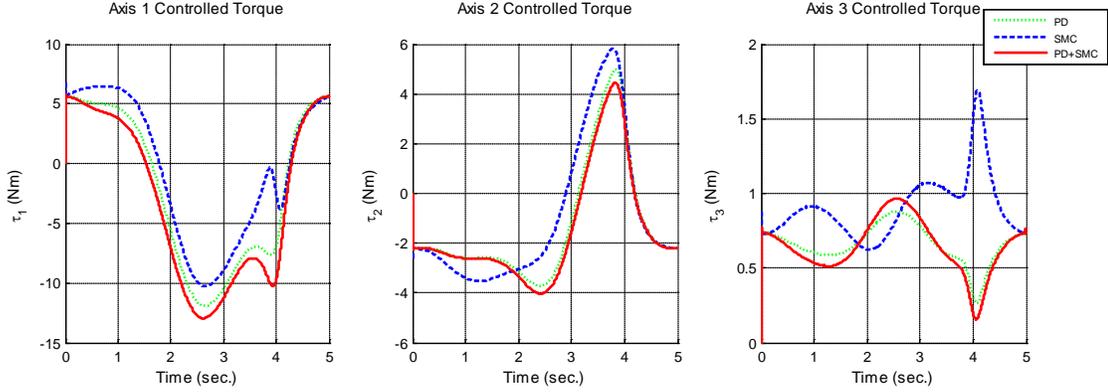


Figure 4.51: Input Torques for Circular Motion of Serial Manipulator

4.3.4 Elliptical Contour

For the elliptical contour simulations, the results listed in Table 4.29 indicate good axial tracking performance for each controller. Once again the PD-SMC produced smaller mean errors than the PD controller in axis 1 and axis 3 by 34.3% and 23.3%, respectively, but the SMC controller produced the smallest mean errors in these axes by 43.1% and 19.7%, respectively, compared to the PD-SMC controller. However, the PD-SMC controller once again produced smallest mean errors in axis 2 by 13.7% and 34.7% relative to the PD and SMC controllers, respectively. The axial tracking errors for each of the controllers are displayed in Figure 4.52.

Table 4.29: Mean and Standard Deviation of Axial Tracking Errors for Elliptical Motion of Serial Manipulator

	Axis 1		Axis 2		Axis 3	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.0006194	0.0003138	0.0002745	0.0001387	0.0002459	0.0001145
SMC	0.0002317	0.0001030	0.0003624	0.0001939	0.0001514	6.053(10 ⁻⁵)
PD-SMC	0.0004070	0.0002056	0.0002368	0.0001192	0.0001885	8.571(10 ⁻⁵)

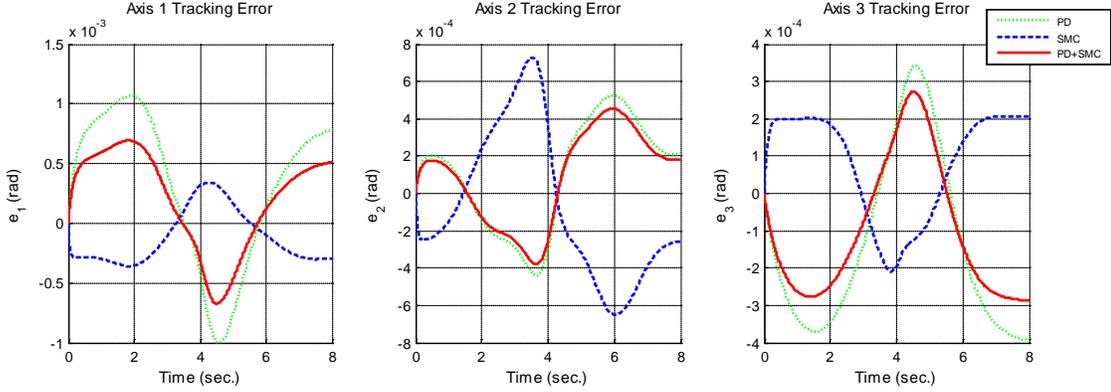


Figure 4.52: Axial Tracking Error for Elliptical Motion of Serial Manipulator

The PD-SMC controller demonstrated a better performance in task space tracking errors than the standard PD controller in each axis as well as with total error magnitudes. It produced a 29.3% lower mean total error magnitude than the PD controller and a 41.5% smaller standard deviation in these errors. The SMC controller once again yielded the best performance in these fields with a smaller mean total tracking error than the PD-SMC controller by 12.8%. The task space tracking results for the elliptical contour can be seen in Figure 4.53 and Table 4.30.

Table 4.30: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Elliptical Motion of Serial Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(<i>m</i>)	S.D. (<i>m</i>)	Mean(<i>m</i>)	S.D. (<i>m</i>)	Mean(<i>m</i>)	S.D. (<i>m</i>)
PD	0.0001789	0.0001285	0.0001968	$7.179(10^{-5})$	0.0002830	0.0001111
SMC	$8.327(10^{-5})$	$4.431(10^{-5})$	0.0001455	$4.355(10^{-5})$	0.0001744	$3.946(10^{-5})$
PD-SMC	0.0001199	$7.441(10^{-5})$	0.0001498	$4.358(10^{-5})$	0.0002001	$6.492(10^{-5})$

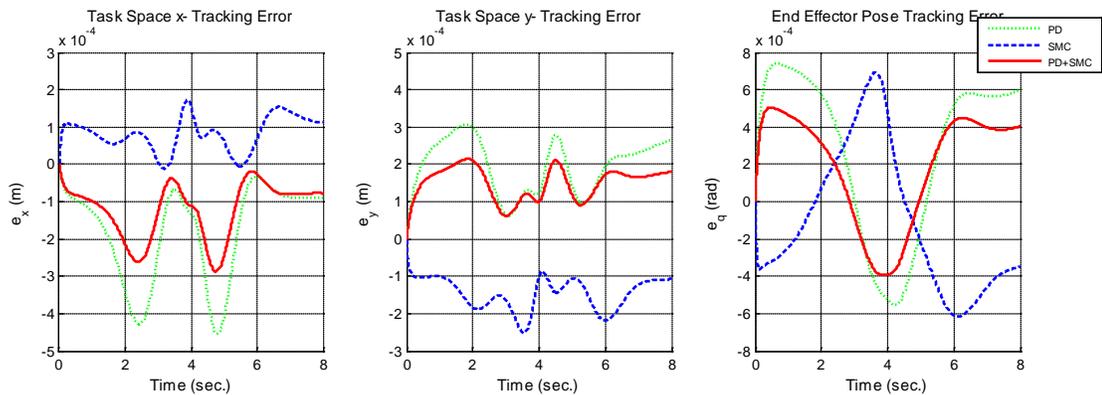


Figure 4.53: End-Effector Task Space Tracking Error for Elliptical Motion of the Serial Manipulator

In terms of contour tracking performance, the PD-SMC once again demonstrates the best performance. The hybrid controller produced a 27.8% smaller mean contour error magnitude than the PD controller and a 48% smaller standard deviation of these errors. Compared to the SMC controller, the PD-SMC controller produced only a marginally improved mean contour error magnitude that was 3.6% smaller, but also yielded a 40.9% standard deviation of these errors. The PD-SMC controller yielded 100% of its contour errors within a boundary of $2(10^{-4}) [m]$ for the performance of this elliptical contour. The SMC controller also yields 100% of contour errors within this boundary while the PD controller yields only 84.6%. The contour tracking results can be seen in Table 4.31 and Figure 4.54, while plots of the desired and actual contours are displayed in Figure 4.55.

Table 4.31: Magnitude of Contour Tracking Performance for Elliptical Motion of Serial Manipulator

	Mean (m)	S.D. (m)	%($\epsilon \leq 2(10^{-4}) [m]$)
PD	0.0001313	$5.590(10^{-5})$	84.64%
SMC	$9.830(10^{-5})$	$4.918(10^{-5})$	100%
PD-SMC	$9.480(10^{-5})$	$2.906(10^{-5})$	100%

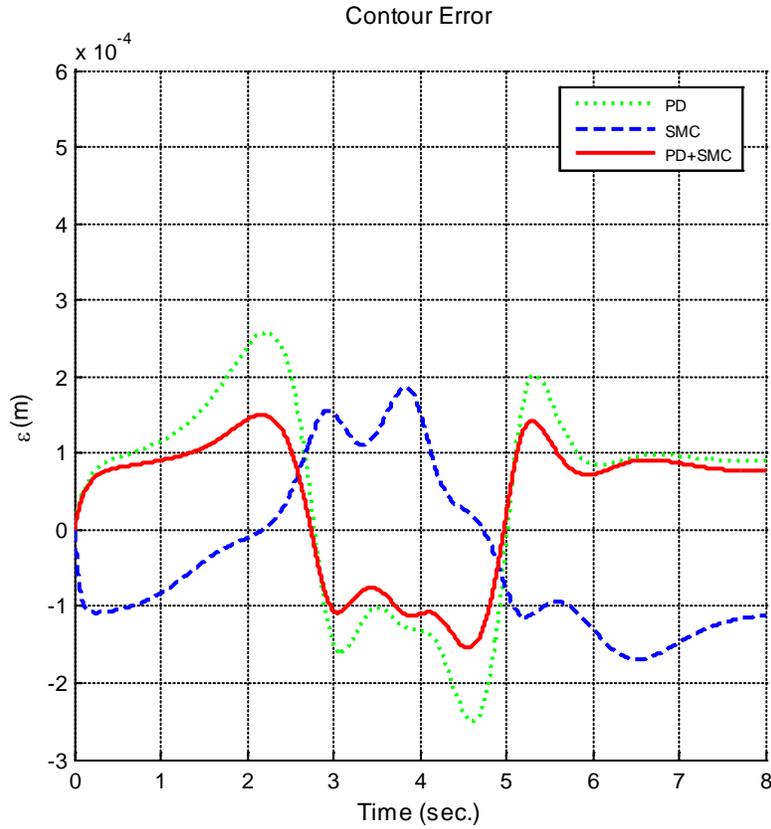


Figure 4.54: Contour Error for Elliptical Motion of the Serial Manipulator

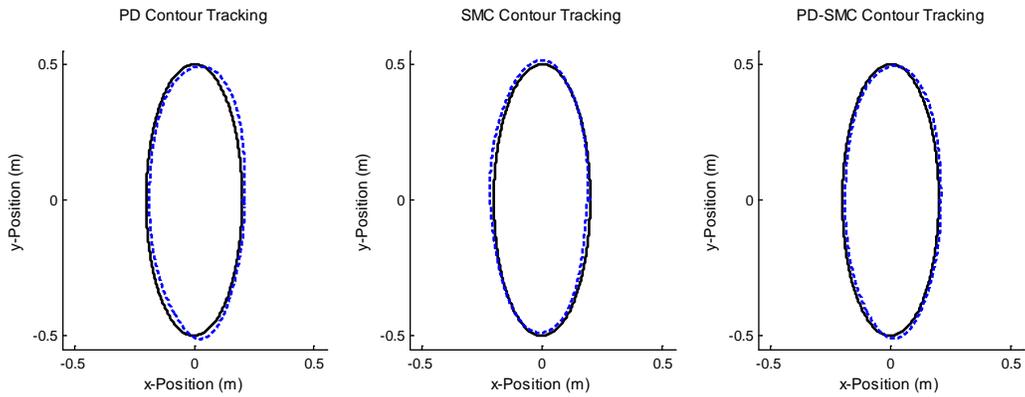


Figure 4.55: Elliptical Contour Tracking of PD, SMC, and PD-SMC Controllers for Serial Manipulator⁴

As displayed in Figure 4.56 the input torques for all the controllers for the elliptical contour are quite similar. Table 4.32 also shows that the maximum torques exerted in each axis for every controller are similar. The largest deviation between any two corresponding maximum

⁴ Plots include an error magnification factor of 100

torque value between any of the controllers is less than 10%. The PD-SMC controller, however, does have a marginally smaller maximum torque output in axis 2 and axis 3 than the other two controllers. For axis 1, the SMC controller produces the lowest maximum torque.

Table 4.32: Maximum Input Torques for Elliptical Motion of Serial Manipulator

	τ_1 (Nm)	τ_2 (Nm)	τ_3 (Nm)
PD	9.455	4.267	0.8885
SMC	9.085	4.370	0.8859
PD-SMC	9.711	4.256	0.8840

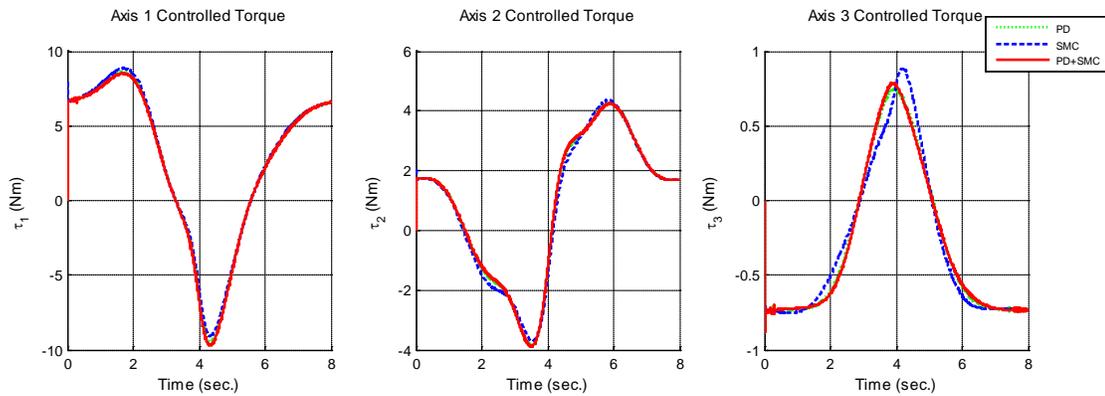


Figure 4.56: Input Torques for Elliptical Motion of Serial Manipulator

4.4 PARALLEL MANIPULATOR SIMULATION RESULTS

The control parameters used for the serial manipulator in the following simulations are listed in Table 4.33. The boundary layer thicknesses used for the SMC and the hybrid PD-SMC controllers are also listed. For the SMC simulations, the uncertainties of the dynamic models are applied by using the following estimations of parameter matrices:

$$\widehat{M}(q') = 0.9\overline{M}(q')$$

$$\widehat{C}(q', \dot{q}') = 0.9C(q', \dot{q}')$$

$$\widehat{G}(q') = 1.1G(q')$$

Table 4.33: Control Parameters for Parallel Manipulator Simulations

Controller Type	Linear Contours	Circular and Horizontal Elliptical Contours	Vertical Elliptical Contour
PD	$K_p = \text{diag}\{220,240\}$	$K_p = \text{diag}\{230,260\}$	$K_p = \text{diag}\{240,280\}$
	$K_d = \text{diag}\{22,35\}$	$K_d = \text{diag}\{24,36\}$	$K_d = \text{diag}\{28,36\}$
SMC	$K_s = \text{diag}\{1,2.2\}$	$K_s = \text{diag}\{1.1,2.25\}$	$K_s = \text{diag}\{1.4,2.4\}$
	$\lambda = \text{diag}\{12,15\}$	$\lambda = \text{diag}\{11,14\}$	$\lambda = \text{diag}\{18,16\}$
	$\phi = \text{diag}\{0.4,0.5\}$	$\phi = \text{diag}\{0.4,0.5\}$	$\phi = \text{diag}\{0.4,0.5\}$
PD-SMC	$K_p = \text{diag}\{220,240\}$	$K_p = \text{diag}\{230,260\}$	$K_p = \text{diag}\{250,280\}$
	$K_d = \text{diag}\{22,35\}$	$K_d = \text{diag}\{24,36\}$	$K_d = \text{diag}\{32,36\}$
	$K_s = \text{diag}\{1,2.2\}$	$K_s = \text{diag}\{1.1,2.25\}$	$K_s = \text{diag}\{1.1,1.4\}$
	$\lambda = \text{diag}\{12,15\}$	$\lambda = \text{diag}\{11,14\}$	$\lambda = \text{diag}\{16,18\}$
	$\phi = \text{diag}\{0.4,0.5\}$	$\phi = \text{diag}\{0.4,0.5\}$	$\phi = \text{diag}\{0.4,0.5\}$

4.4.1 Zigzag Contour

It should be noted that only the errors for the controlled axes of the parallel manipulator are listed for the simulations. From the results shown in Table 4.34 and displayed in Figure 4.57 it is seen that the PD-SMC controller achieved the best performance by producing the lowest mean axial tracking error in axis 1, while being outperformed only marginally by the SMC controller in axis 2 for this field. The PD-SMC controller yielded smaller mean axial errors than the PD controller by 11.9% (axis 1) and 21.4% (axis 2). It outperformed the SMC controller in this field by 35.7% (axis 1).

The PD-SMC further outperformed the other PD and SMC controllers in the task space by producing a smaller total tracking error mean by 18.2% and 15.9%, respectively. It also yielded smaller standard deviations than the PD and SMC for these errors by 14.8% and 26.6%. The task space results are shown in Table 4.35 and Figure 4.58.

Table 4.34: Mean and Standard Deviation of Axial Tracking Errors for Zigzag Motion of Parallel Manipulator

	Axis 1		Axis 2	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.001490	0.001490	0.002032	0.0002624
SMC	0.002040	0.0005700	0.001490	0.0001817
PD-SMC	0.001313	0.0004215	0.001597	0.0001981

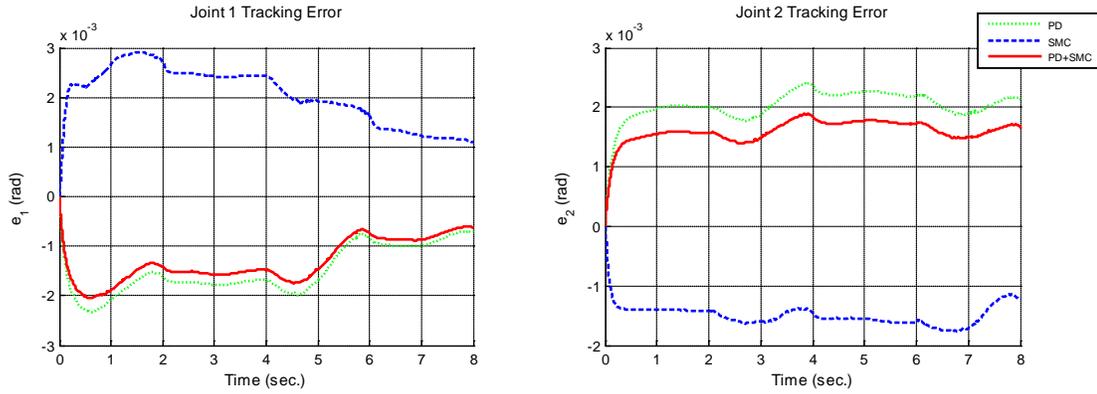


Figure 4.57: Axial Tracking Error for Zigzag Motion of Parallel Manipulator

Table 4.35: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Zigzag Motion of Parallel Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	2.917(10 ⁻⁵)	2.087(10 ⁻⁵)	0.0002336	4.501(10 ⁻⁵)	0.000236362	4.494(10 ⁻⁵)
SM	3.391(10 ⁻⁵)	1.901(10 ⁻⁵)	0.0002265	5.235(10 ⁻⁵)	0.000229879	5.215(10 ⁻⁵)
C						
PD-SM	1.933(10 ⁻⁵)	1.985(10 ⁻⁵)	0.0001919	3.843(10 ⁻⁵)	0.000193436	3.830(10 ⁻⁵)

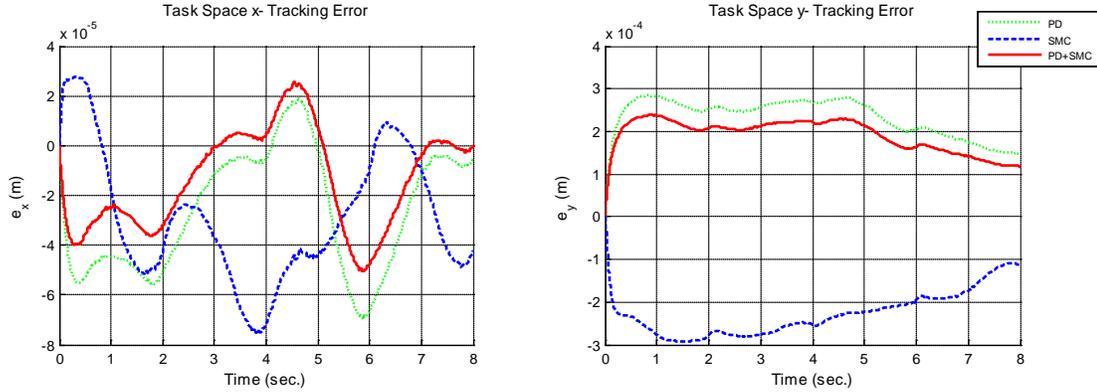


Figure 4.58: End-Effector Task Space Tracking Error for Zigzag Motion of the Parallel Manipulator

In terms of contour tracking performance the PD-SMC once again achieved better results by producing a smaller mean magnitude of contour error by 18.1% and 13.8% than the PD and SMC controllers, respectively. The contour tracking results are summarized in Table 4.36. From these results it can be seen that for the performance of this contour, 79.0% of the contour errors produced by the PD-SMC controller are smaller than $1.5(10^{-4}) [m]$. The PD and SMC controllers only achieve 64.4% and 41.35% of this statistic, respectively. The contour error performance of the three controllers is displayed in Figure 4.59, and the desired and actual contours can be seen in Figure 4.60.

Table 4.36: Magnitude of Contour Tracking Performance for Zigzag Motion of Parallel Manipulator

	Mean (m)	S.D. (m)	$\%(\varepsilon \leq 1.5(10^{-4}) [m])$
PD	0.0001449	$4.201(10^{-5})$	41.35%
SMC	0.0001376	$3.916(10^{-5})$	64.39%
PD-SMC	0.0001186	$3.112(10^{-5})$	78.95%

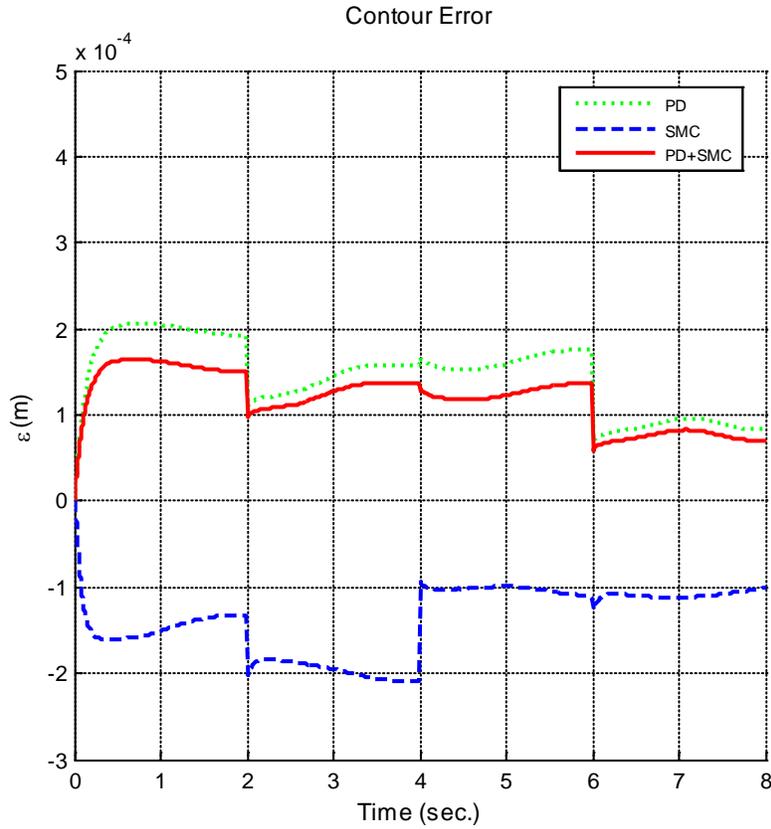


Figure 4.59: Contour Error for Zigzag Motion of the Parallel Manipulator

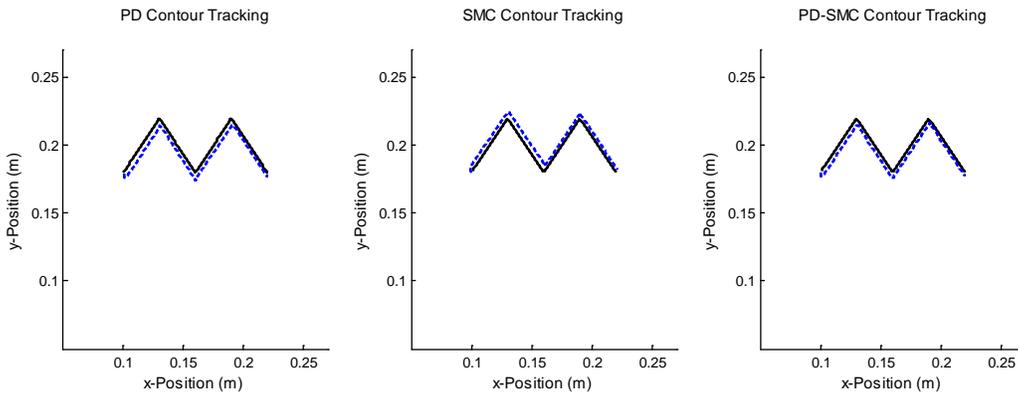


Figure 4.60: Zigzag Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator⁵

The maximum torque data, listed in Table 4.37, indicates that the PD controller requires the greatest maximum torque for axis 2 for the zigzag contour, while the PD-SMC controller requires the largest torque for axis 1. While there are differences in the required torque, the

⁵ Plots include an error magnification factor of 20

variance is by less than 1% in both axes. The input torques for the controlled axes are shown in Figure 4.61 and it can be seen that the torque requirements are similar for all three controllers.

Table 4.37: Maximum Input Torques for Zigzag Motion of Parallel Manipulator

	τ_1 (Nm)	τ_2 (Nm)
PD	0.5111	0.5768
SMC	0.5083	0.5739
PD-SMC	0.5155	0.5739

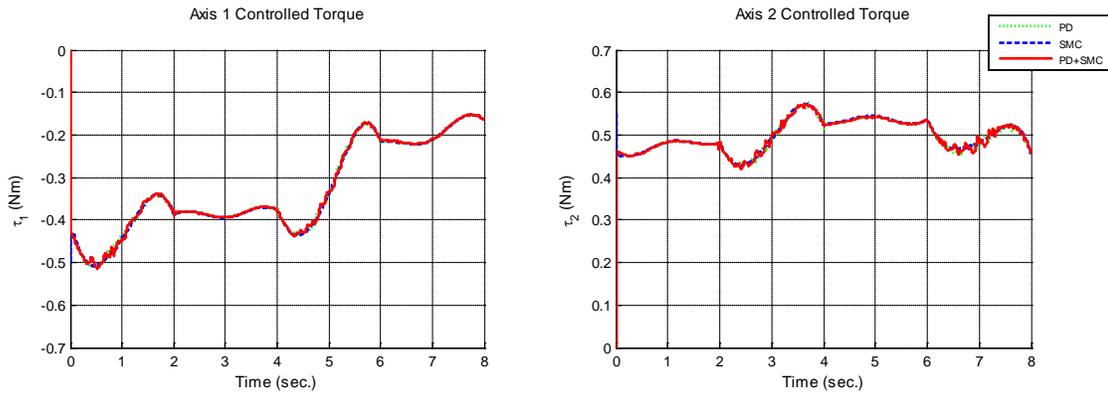


Figure 4.61: Input Torques for Zigzag Motion of Parallel Manipulator

4.4.2 *Diamond Contour*

The PD-SMC controller once again yielded smaller mean axial tracking errors than the PD controller in each axis. Relative to the SMC controller, the PD-SMC controller maintained a better performance in axis 1. The axial tracking results are listed in Table 4.38 and displayed in Figure 4.62.

The PD-SMC controller maintained a better performance in terms of task space tracking errors by yielding a smaller mean than the PD and SMC controllers by 18.4% and 14.2%, respectively. The task space tracking results are listed in Table 4.39 and displayed in Figure 4.63.

Table 4.38: Mean and Standard Deviation of Axial Tracking Errors for Diamond Motion of Parallel Manipulator

	Axis 1		Axis 2	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.001314	0.0006642	0.001936	0.0003280
SMC	0.001856	0.0005832	0.001389	0.0002092
PD-SMC	0.001160	0.0005836	0.001519	0.0002502

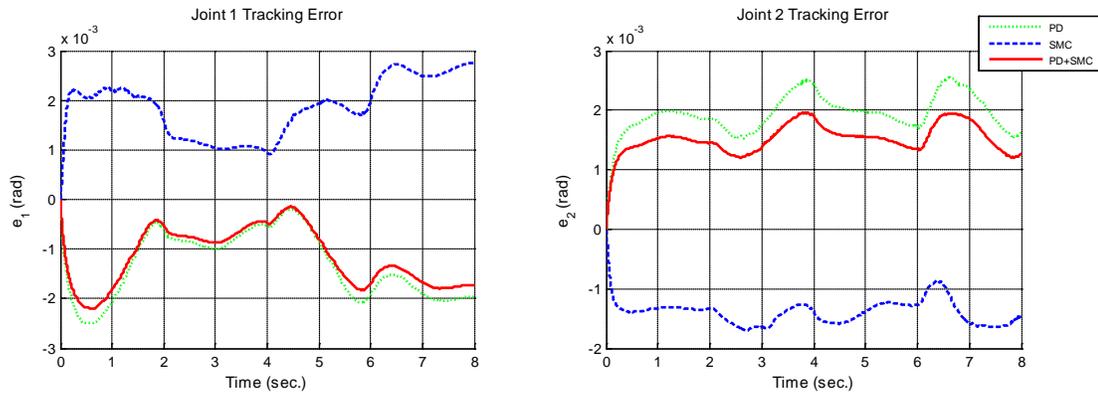


Figure 4.62: Axial Tracking Error for Diamond Motion of Parallel Manipulator

Table 4.39: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Diamond Motion of Parallel Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	$4.859(10^{-5})$	$2.464(10^{-5})$	0.0002004	$4.976(10^{-5})$	0.0002084	$4.655(10^{-5})$
SMC	$3.714(10^{-5})$	$3.019(10^{-5})$	0.0001919	$4.791(10^{-5})$	0.0001981	$4.670(10^{-5})$
PD-SMC	$3.556(10^{-5})$	$3.571(10^{-5})$	0.0001643	$4.257(10^{-5})$	0.0001700	$3.972(10^{-5})$

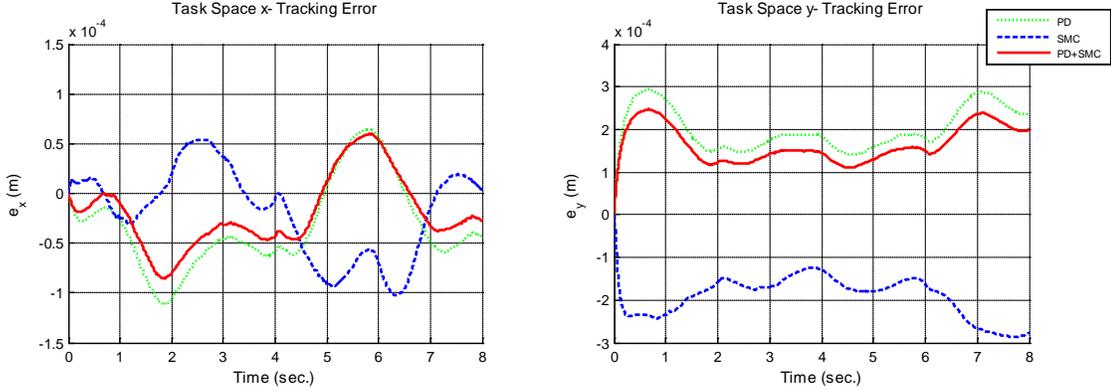


Figure 4.63: End-Effector Task Space Tracking Error for Diamond Motion of the Parallel Manipulator

In terms of contour tracking performance, the PD-SMC controller achieved better results than the other two controllers by producing a smaller mean contour error magnitude and standard deviation of the contour errors. Compared to the PD controller, the PD-SMC yielded a 17.9% smaller mean contour error magnitude and 20.1% smaller standard deviation of these errors. The PD-SMC controller also produced an 11.8% smaller mean contour error than the SMC and a 23.1% smaller standard deviation. The amount of contour errors that fall within a $1.5(10^{-4}) [m]$ error boundary for the PD controller is 55.5% while the SMC and PD-SMC yield 64.4% and 83.8% for the same statistic, respectively. The contour error performance of the controllers is shown in Figure 4.64 while the desired and actual plots of the contour are shown in Figure 4.65.

Table 4.40: Magnitude of Contour Tracking Performance for Diamond Motion of Parallel Manipulator

	Mean (<i>m</i>)	S.D. (<i>m</i>)	%(ϵ $\leq 1.5(10^{-4}) [m]$)
PD	0.0001350	$4.922(10^{-5})$	55.52%
SMC	0.0001257	$5.101(10^{-5})$	64.35%
PD-SMC	0.0001108	$3.923(10^{-5})$	83.79%

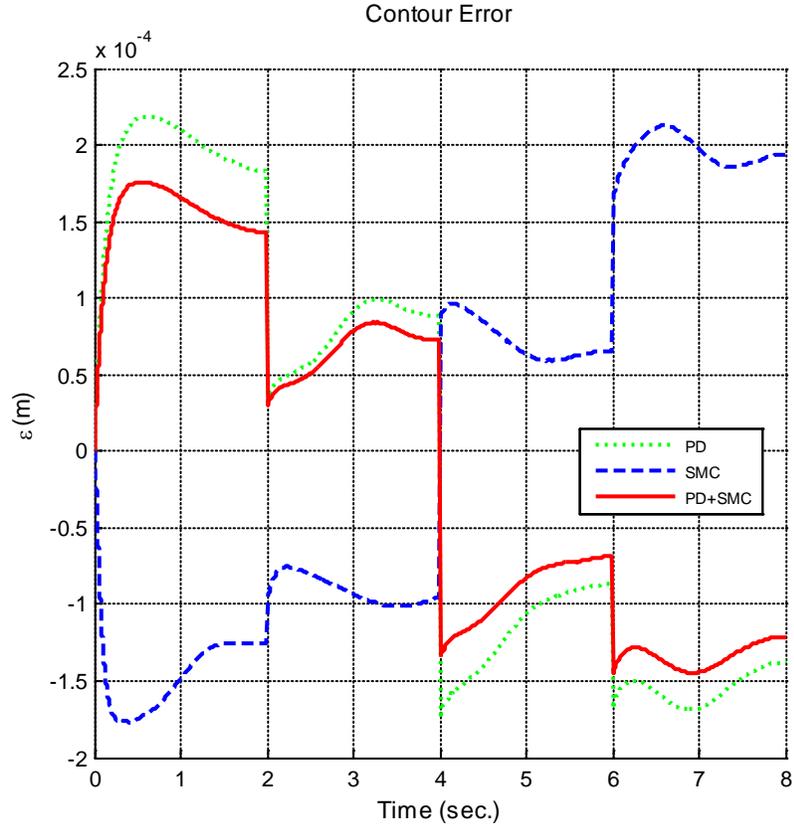


Figure 4.64: Contour Error for Diamond Motion of the Parallel Manipulator

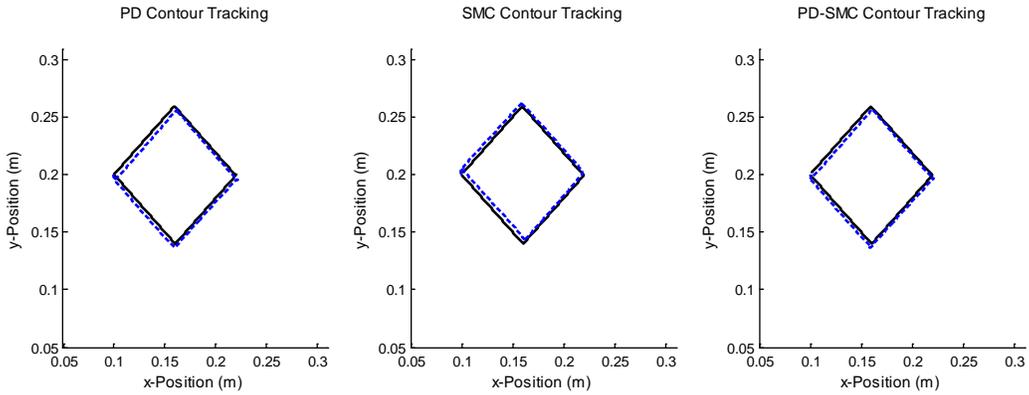


Figure 4.65: Diamond Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator⁶

Table 4.41: Maximum Input Torques for Diamond Motion of Parallel Manipulator

	τ_1 (Nm)	τ_2 (Nm)
PD	0.5655	0.6155
SMC	0.5595	0.6107
PD-SMC	0.5556	0.6022

⁶ Plots include an error magnification factor of 20

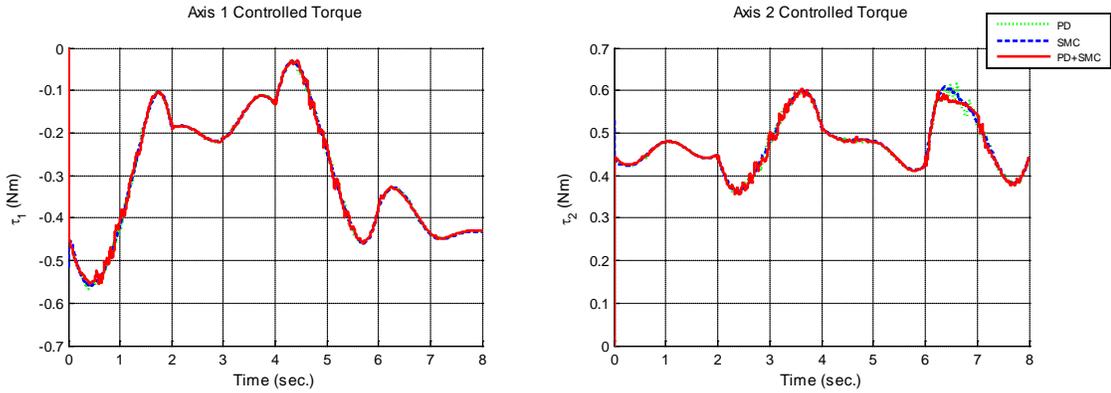


Figure 4.66: Input Torques for Diamond Motion of Parallel Manipulator

From the maximum torque results listed in Table 4.41, it is shown that the PD-SMC yielded the smallest maximum torque requirement for both controlled axes. The SMC yielded the highest maximum torque for axis 2, while the PD yielded the highest maximum torque for axis 1. Input torque performance for the diamond contour is displayed in Figure 4.66

4.4.3 *Circular Contour*

For the circular contour, the PD-SMC achieved the best axial tracking performance by producing the smallest mean axial tracking errors and standard deviations in each of the four axes. From Table 4.42 it can be seen that in terms of mean axial tracking error, the PD-SMC outperformed the PD controller by 10% – 20%. Compared to the SMC controller, the PD-SMC performed marginally better in axis 2 and more than 40% better in axis 1.

In terms of task space tracking error, the PD-SMC once again performed better than the PD and SMC controllers by yielding a smaller mean total tracking error by 17.5% and 16.1%, respectively. These results are listed in Table 4.43 while the task space tracking error performance can be seen in Figure 4.68.

Table 4.42: Mean and Standard Deviation of Axial Tracking Errors for Circular Motion of Parallel Manipulator

	Axis 1		Axis 2	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.0009260	0.0006140	0.001791	0.0002711
SMC	0.001365	0.0006702	0.001453	0.0002484
PD-SMC	0.0008174	0.0005419	0.001445	0.0002134

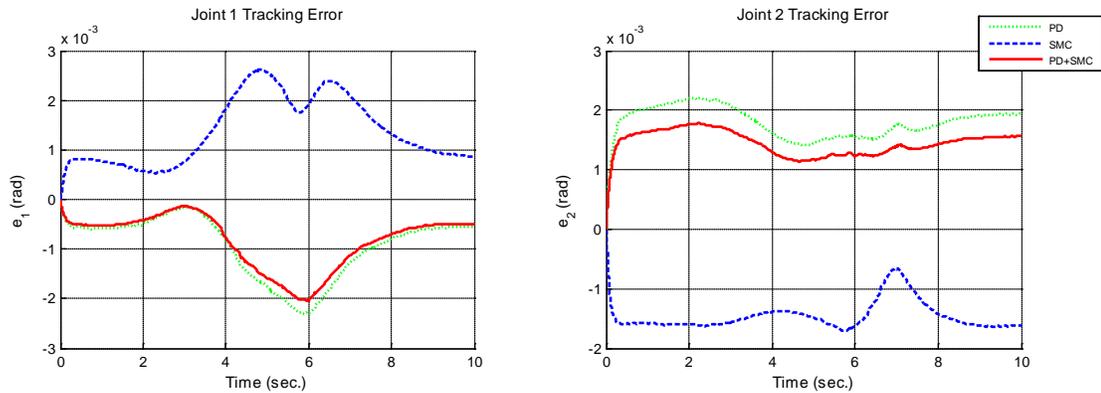


Figure 4.67: Axial Tracking Error for Circular Motion of Parallel Manipulator

Table 4.43: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Circular Motion of Parallel Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	$5.511(10^{-5})$	$3.547(10^{-5})$	0.0001602	$3.192(10^{-5})$	0.0001728	$3.356(10^{-5})$
SMC	$4.175(10^{-5})$	$3.624(10^{-5})$	0.0001597	$4.408(10^{-5})$	0.0001697	$4.110(10^{-5})$
PD-SMC	$4.353(10^{-5})$	$4.581(10^{-5})$	0.0001326	$2.743(10^{-5})$	0.0001425	$2.757(10^{-5})$

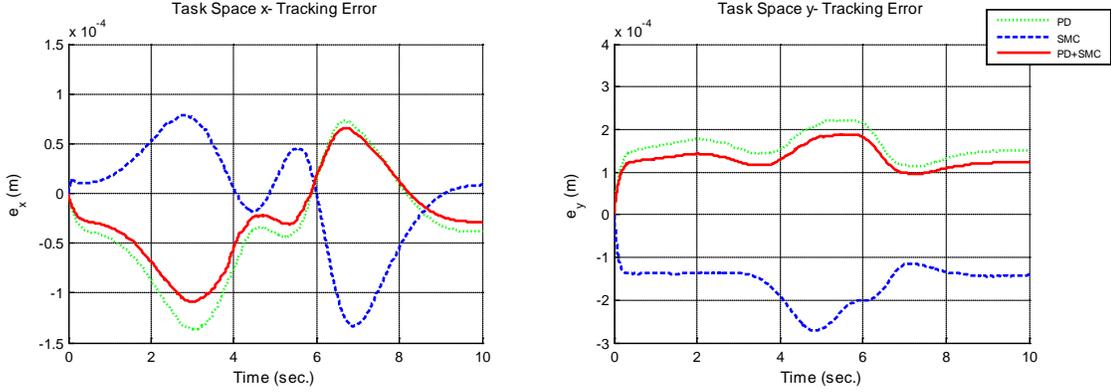


Figure 4.68: End-Effector Task Space Tracking Error for Circular Motion of the Parallel Manipulator

From the results of the contour tracking performance listed in Table 4.44, the SMC controller is shown to have yielded the smallest magnitude of mean contour error. The PD-SMC controller has a smaller mean than the PD controller by 19.3%, but the SMC has a smaller mean than the PD-SMC by 3.4%. Despite having a smaller mean contour error, the SMC controller also yielded the largest standard deviation of these errors. The standard deviation for the PD controller is 14.2% smaller than that of the SMC while the PD-SMC produced a smaller standard deviation than the SMC by 26.6%. Despite the smaller standard deviation of contour errors of the PD controller, fewer contour error magnitudes are within a $1.5(10^{-4})[m]$ boundary of error for the SMC controller, as shown in Table 4.44. The PD-SMC controller yields the most amount of the motion within the error boundary. The contour errors and actual contour plots are shown in Figure 4.69 and Figure 4.70, respectively.

Table 4.44: Magnitude of Contour Tracking Performance for Circular Motion of Parallel Manipulator

	Mean (m)	S.D. (m)	$\%(\epsilon \leq 1.5(10^{-4}) [m])$
PD	$6.958(10^{-5})$	$5.533(10^{-5})$	84.86%
SMC	$5.433(10^{-5})$	$6.451(10^{-5})$	85.04%
PD-SMC	$5.616(10^{-5})$	$4.736(10^{-5})$	94.46%

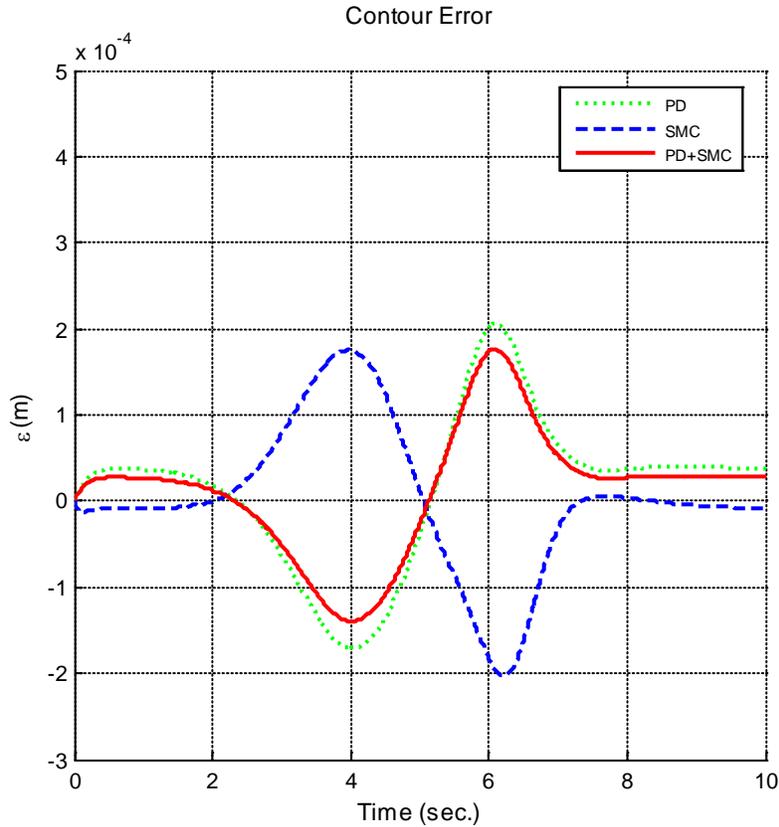


Figure 4.69: Contour Error for Circular Motion of the Parallel Manipulator

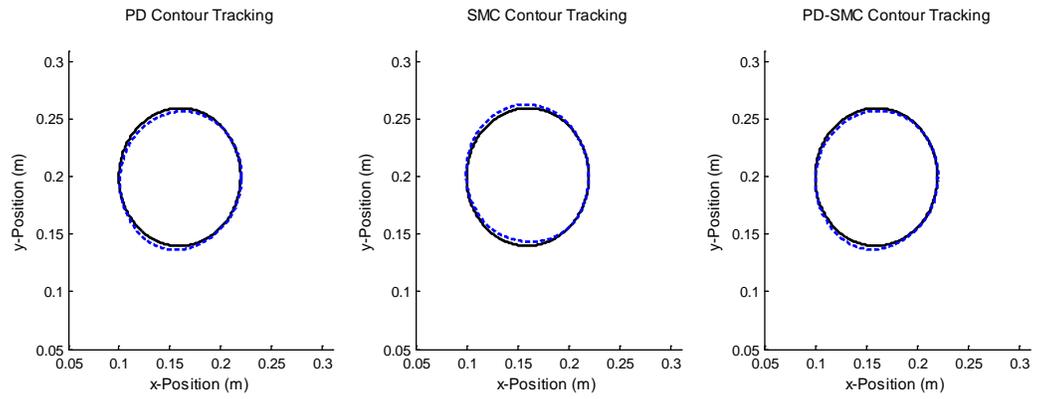


Figure 4.70: Circular Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator⁷

From Table 4.45 it is seen that the SMC controller had the largest maximum torques for axis 2, while the PD-SMC controller had the largest maximum torques for axis 1. The difference in maximum torque for axis 1 was smaller than 1%, while the PD and PD-SMC controllers

⁷ Plots include an error magnification factor of 20

yielded smaller maximum torques than the SMC controller by more than 5%. The torque performance of each controlled axis is displayed in Figure 4.71.

Table 4.45: Maximum Input Torques for Circular Motion of Parallel Manipulator

	τ_1 (Nm)	τ_2 (Nm)
PD	0.5337	0.5754
SMC	0.5278	0.6172
PD-SMC	0.5340	0.5806

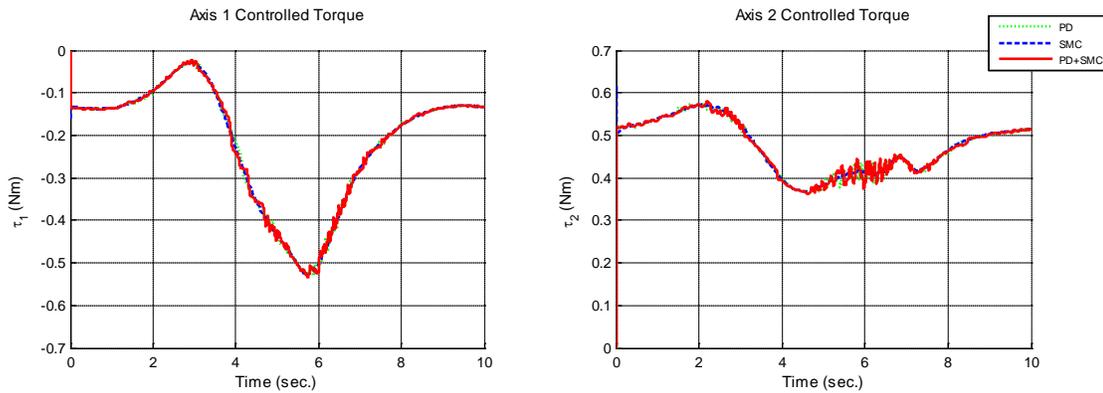


Figure 4.71: Input Torques for Circular Motion of Parallel Manipulator

4.4.4 Horizontal Elliptical Contour

Similar to the circular contour results, the PD-SMC produced the smallest mean axial tracking errors in each axis for the horizontal elliptical contours, despite producing larger standard deviations of this error than the SMC controller. The axial tracking error mean and standard deviation results are listed in Table 4.46 while the axial performance results are displayed in Figure 4.72.

As shown in Table 4.47, the PD-SMC achieved a better end-effector task space tracking error mean than both the PD and SMC controller by about 15%. The SMC produced the largest

standard deviation of these errors for this contour. The task space tracking results are shown in Figure 4.71.

Table 4.46: Mean and Standard Deviation of Axial Tracking Errors for Horizontal Elliptical Motion of Parallel Manipulator

	Axis 1		Axis 2	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.001048	0.0007373	0.001913	0.0003724
SMC	0.001528	0.0005872	0.001562	0.0001808
PD-SMC	0.000929	0.0006477	0.001541	0.0002963

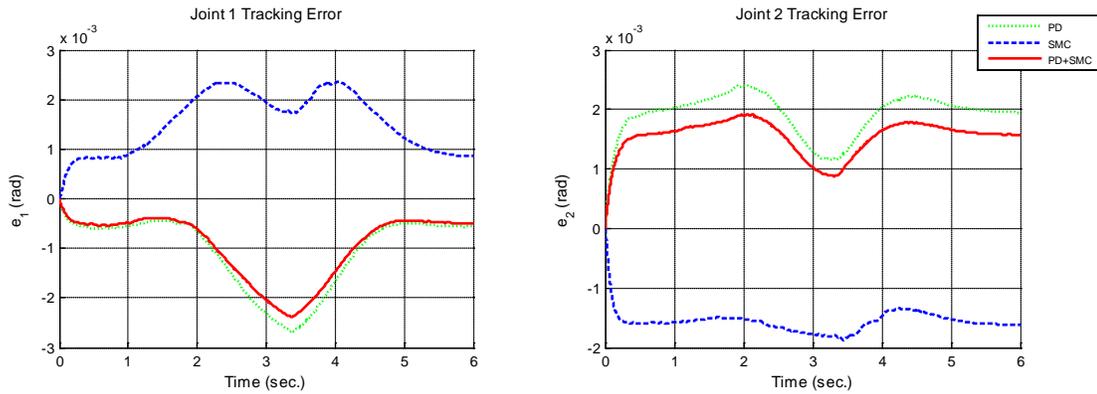


Figure 4.72: Axial Tracking Error for Horizontal Elliptical Motion of Parallel Manipulator

Table 4.47: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Horizontal Elliptical Motion of Parallel Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	$4.018(10^{-5})$	$1.994(10^{-5})$	0.0001926	$4.786(10^{-5})$	0.0001978	$4.750(10^{-5})$
SMC	$2.996(10^{-5})$	$2.527(10^{-5})$	0.0001895	$5.483(10^{-5})$	0.0001931	$5.637(10^{-5})$
PD-SMC	$3.111(10^{-5})$	$2.638(10^{-5})$	0.0001598	$4.160(10^{-5})$	0.0001634	$4.132(10^{-5})$

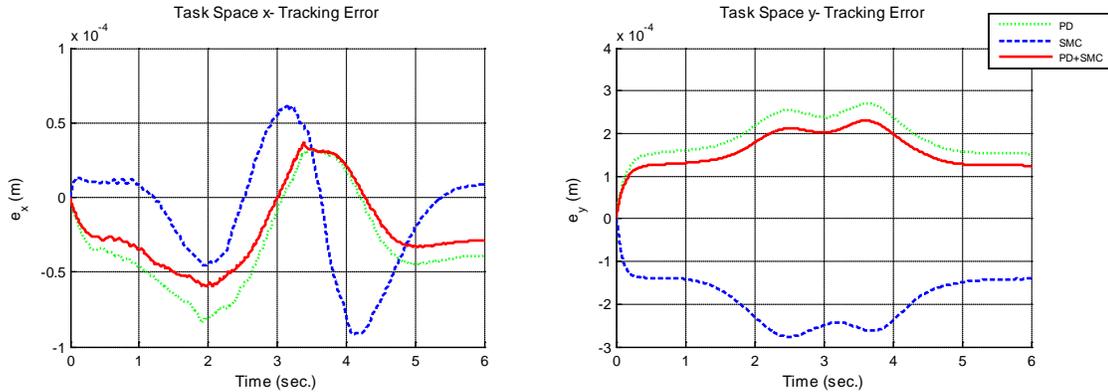


Figure 4.73: End-Effector Task Space Tracking Error for Horizontal Elliptical Motion of the Parallel Manipulator

Moderate contour performance for the horizontal ellipse was achieved by each of the controllers using the controller gains listed in Table 4.33. The PD-SMC controller, however, remained the best performer of the three controllers by achieving the smallest mean contour error magnitude and standard deviation, as listed in Table 4.48. Again, the PD-SMC controller yielded the most contour errors to fall within an error boundary of $1.5(10^{-4})$ [m] for the performance of the horizontal elliptical. The contour performance of each of the controllers is displayed in Figure 4.74 while the desired and actual plots of the contour are displayed in Figure 4.75.

Table 4.48: Magnitude of Contour Tracking Performance for Horizontal Elliptical Motion of Parallel Manipulator

	Mean (m)	S.D. (m)	$\%(\epsilon \leq 1.5(10^{-4})$ [m])
PD	0.0001560	$8.294(10^{-5})$	42.14%
SMC	0.0001460	$9.190(10^{-5})$	49.18%
PD-SMC	0.0001290	$7.051(10^{-5})$	57.47%

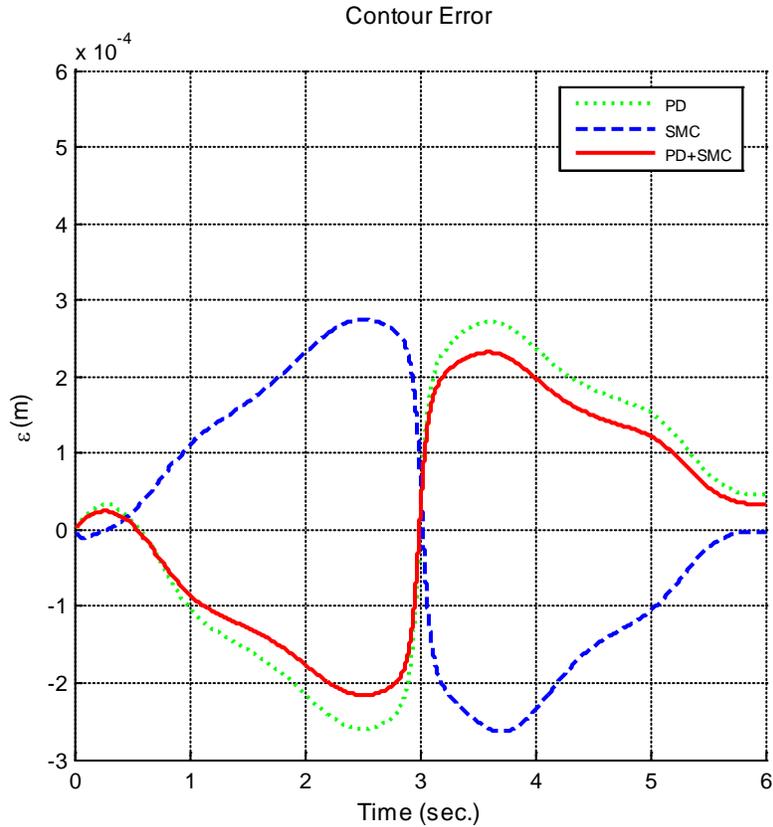


Figure 4.74: Contour Error for Horizontal Elliptical Motion of the Parallel Manipulator

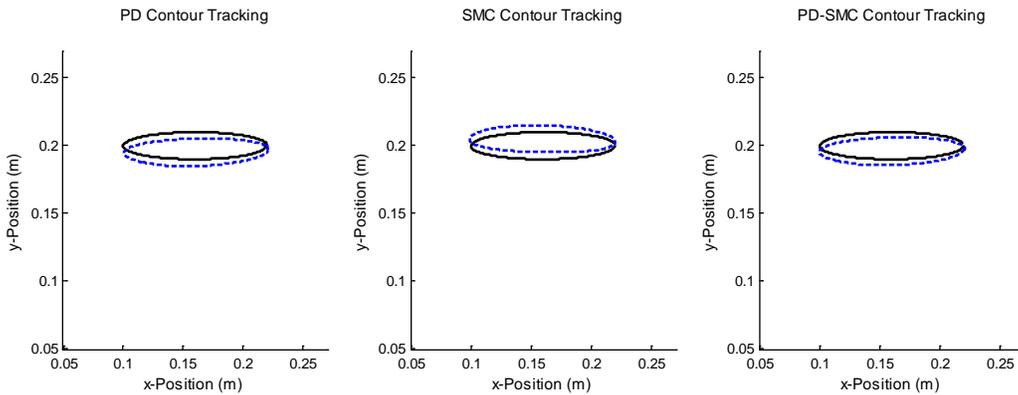


Figure 4.75: Horizontal Elliptical Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator⁸

The SMC controller produced the smallest maximum torque requirements for each axis for the horizontal elliptical contour. From Table 4.49 the difference in maximum torque requirements between the PD and PD-SMC controllers appears only marginal while the

⁸ Plots include an error magnification factor of 20

maximum torque of the SMC controller for both axes is more about 2% smaller than those of the PD and SMC. The input torque performance of the controlled axes is displayed in Figure 4.76.

Table 4.49: Maximum Input Torques for Horizontal Elliptical Motion of Parallel Manipulator

	τ_1 (Nm)	τ_2 (Nm)
PD	0.6186	0.6312
SMC	0.6026	0.6201
PD-SMC	0.6187	0.6332

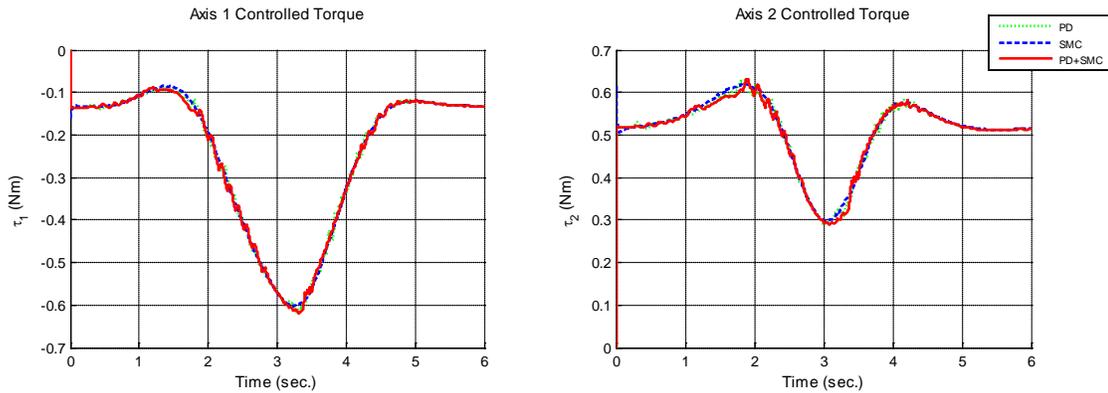


Figure 4.76: Input Torques for Horizontal Elliptical Motion of Parallel Manipulator

4.4.5 Vertical Elliptical Contour

The results of the vertical elliptical contour simulation listed in Table 4.50 show that similar to the results from the other nonlinear contours, the PD-SMC yielded smaller mean axial errors than the other two controllers. The SMC controller achieved a smaller standard deviation of axial tracking error than the PD-SMC controller by 21%, but the PD-SMC controller yielded the smallest standard deviations in each of the other axes. The axial tracking performance of each of the controllers is shown in Figure 4.77.

In addition to the best axial tracking results, the PD-SMC also maintained the best end-effector task space tracking results for each parallel manipulator simulation. From Table 4.51,

the PD-SMC is shown to outperform the SMC controller, by 12.9% and 18.8% in mean end-effector tracking error and standard deviation, respectively.

Table 4.50: Mean and Standard Deviation of Axial Tracking Errors for Vertical Elliptical Motion of Parallel Manipulator

	Axis 1		Axis 2	
	Mean(rad)	S.D. (rad)	Mean(rad)	S.D. (rad)
PD	0.001195	0.0006349	0.001704	0.0005014
SMC	0.001300	0.0004258	0.001478	0.0004725
PD-SMC	0.001023	0.0005392	0.001391	0.0004003

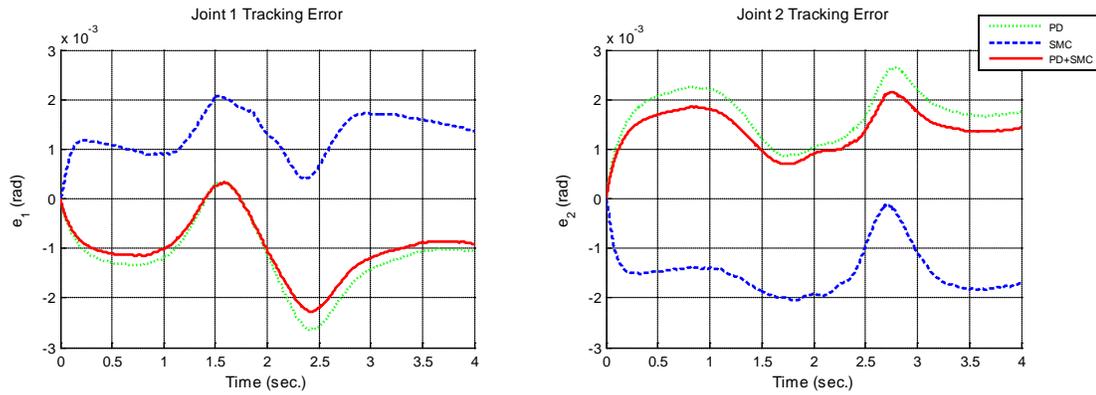


Figure 4.77: Axial Tracking Error for Vertical Elliptical Motion of Parallel Manipulator

Table 4.51: Mean and Standard Deviation of End-Effector Task Space Tracking Errors for Vertical Elliptical Motion of Parallel Manipulator

	x-axis		y-axis		Total Error Magnitude	
	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)	Mean(m)	S.D. (m)
PD	4.100(10 ⁻⁵)	3.290(10 ⁻⁵)	0.0001778	5.277(10 ⁻⁵)	0.0001871	4.624(10 ⁻⁵)
SMC	3.244(10 ⁻⁵)	2.176(10 ⁻⁵)	0.0001734	5.005(10 ⁻⁵)	0.0001785	4.751(10 ⁻⁵)
PD-SMC	3.329(10 ⁻⁵)	4.219(10 ⁻⁵)	0.0001477	4.436(10 ⁻⁵)	0.0001554	3.856(10 ⁻⁵)

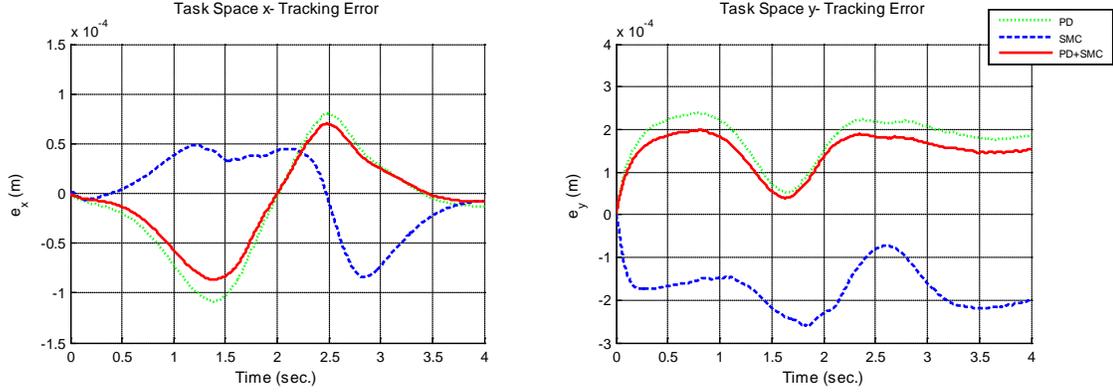


Figure 4.78: End-Effector Task Space Tracking Error for Vertical Elliptical Motion of the Parallel Manipulator

From the results shown in Table 4.52, it can be seen that the SMC and PD-SMC controllers produced similar mean contour error magnitudes and marginally different standard deviations for the vertical elliptical contour. Since the mean contour errors and standard deviations are similar for each controller, there is not a significant deviation when comparing the distribution of the contour errors within the $1.5(10^{-4})$ [m] error boundary. However, the PD-SMC still maintained better contour tracking performance than each other the other controllers. The contour errors of the controllers can be seen in Figure 4.79 while the desired and actual plots are displayed in Figure 4.80.

Table 4.52: Magnitude of Contour Tracking Performance for Vertical Elliptical Motion of Parallel Manipulator

	Mean (m)	S.D. (m)	$\%(\epsilon \leq 1.5(10^{-4})$ [m])
PD	$4.637(10^{-5})$	$5.698(10^{-5})$	92.03%
SMC	$3.760(10^{-5})$	$5.054(10^{-5})$	93.45%
PD-SMC	$3.696(10^{-5})$	$4.875(10^{-5})$	93.98%

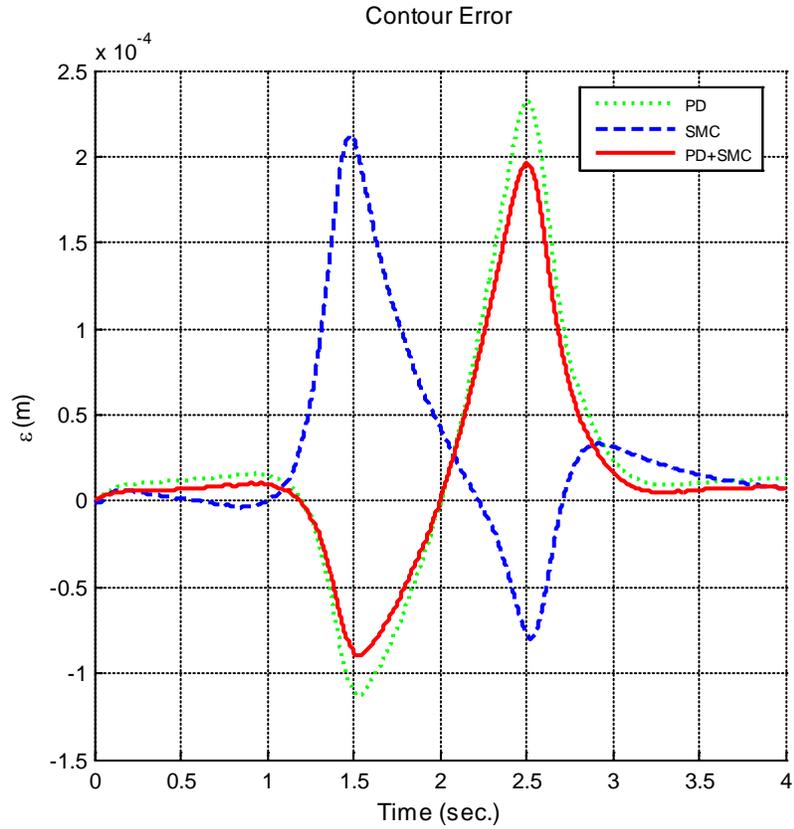


Figure 4.79: Contour Error for Vertical Elliptical Motion of the Parallel Manipulator

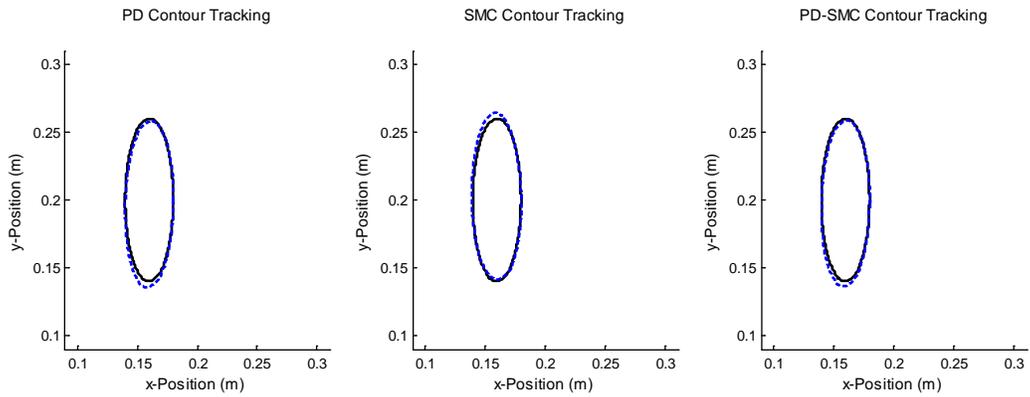


Figure 4.80: Vertical Elliptical Contour Tracking of PD, SMC, and PD-SMC Controllers for Parallel Manipulator⁹

For the vertical elliptical contour the SMC controller had the largest maximum torque input for axis 1 and the smallest maximum torque inputs in axis 2. Again, the maximum torque

⁹ Plots include an error magnification factor of 20

for each controller differs by less than 1%. The torque performance of each controlled axis is displayed in Figure 4.81.

Table 4.53: Maximum Input Torques for Vertical Elliptical Motion of Parallel Manipulator

	τ_1 (Nm)	τ_2 (Nm)
PD	0.7053	0.8021
SMC	0.7077	0.7943
PD-SMC	0.7019	0.8021

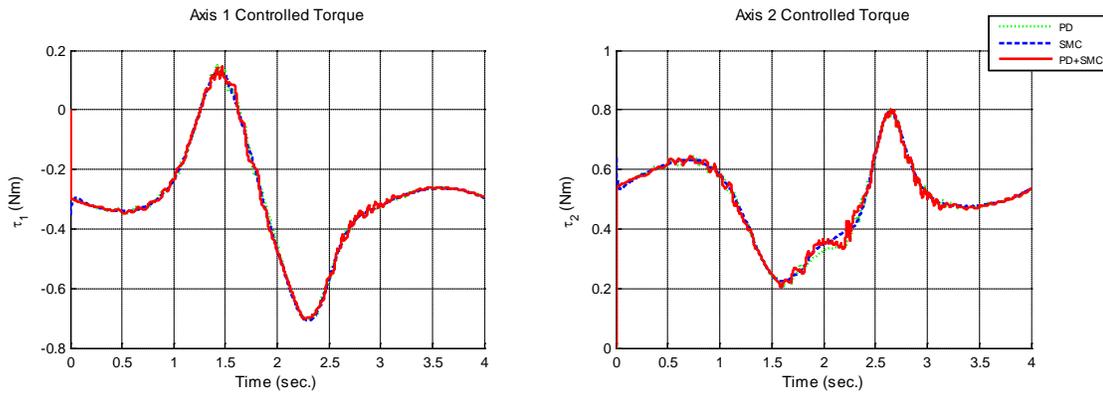


Figure 4.81: Input Torques for Vertical Elliptical Motion of Parallel Manipulator

4.5 OTHER SIMULATIONS

In order to demonstrate the robustness of the proposed controller, simulations with higher-speed movement and simulations with an additional payload at the end-effector were carried out under the same control parameters used in previous simulations. For these simulations, the elliptical contour of the serial manipulator was selected because it was for this contour that the best contour tracking performance was achieved. In addition, the vertical elliptical contour for the parallel manipulator was also simulated since similar performance was exhibited by every controller so the relative robustness of the controllers will be highlighted. Simulations for these contours were also performed with increased gains for which they were

previously tuned in order to further verify the improved performance and robustness of the PD-SMC controller.

4.5.1 High-Speed Movement Simulations

For each of the simulated high-speed cases, the total duration of the movement was reduced to 25% of the originally simulated duration time as outlined in Section 4.2. For the serial manipulator this means that the total duration of the movement was reduced to 2 seconds. The contour tracking performance for the serial manipulator for the high-speed case is listed in Table 4.54 and displayed in Figure 4.82. For the parallel manipulator, the total movement duration was reduced to 1 second. The contour tracking results for the parallel manipulator are shown in Table 4.55 and Figure 4.83.

Table 4.54: Magnitude of Contour Tracking Performance for High-Speed Motion of Serial Manipulator

	Mean (m)	S.D. (m)	$\%(\varepsilon \leq 2(10^{-4}) [m])$
PD	0.0002033	0.0001564	57.81%
SMC	0.0002009	0.0001403	53.63%
PD-SMC	0.0001509	0.0001193	68.25%

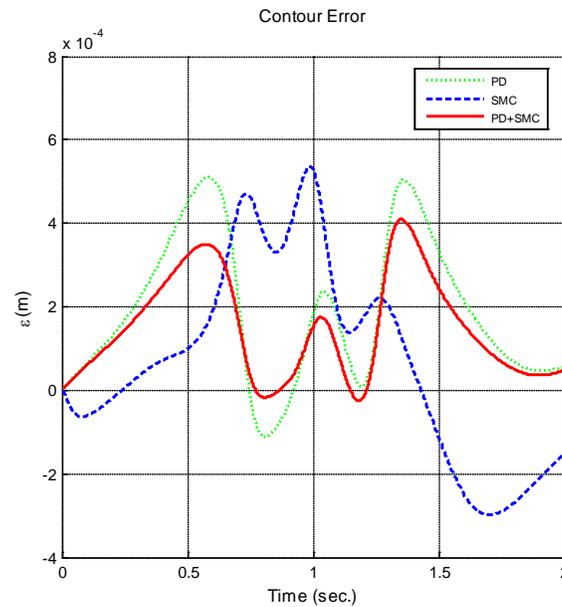


Figure 4.82: Contour Error for High-Speed Motion of the Serial Manipulator

Table 4.55: Magnitude of Contour Tracking Performance for High-Speed Motion of Parallel Manipulator

	Mean (<i>m</i>)	S.D. (<i>m</i>)	%(ϵ \leq $1.5(10^{-4})$ [<i>m</i>])
PD	0.0001363	0.0001947	78.52%
SMC	0.0003430	0.0003320	38.87%
PD-SMC	0.0001266	0.0001756	79.76%

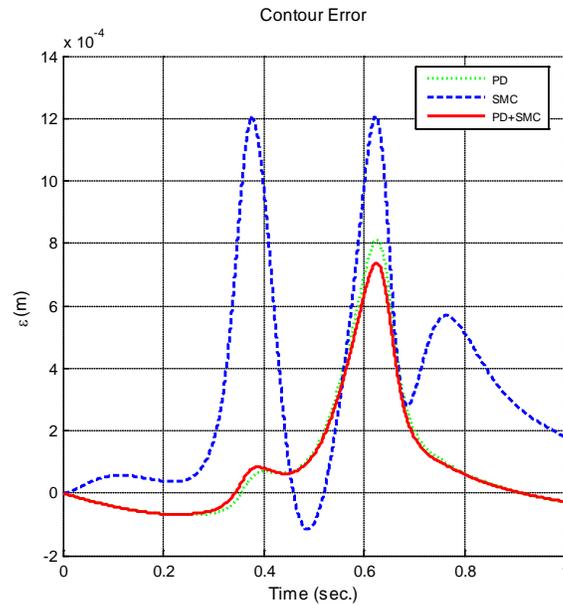


Figure 4.83: Contour Error for High-Speed Motion of the Parallel Manipulator

From the results of the high-speed simulation it is seen that PD-SMC controller maintained the best contour tracking performance of the three controllers, while the quality of the SMC controller's performance was significantly reduced. While there was a 55% increase in the mean contour error magnitude for the serial manipulator with the PD controller and an 60% increase for the PD-SMC controller, the SMC controller yielded a 104% increase in its mean contour error. For the parallel manipulator, the PD and PD-SMC controllers remained similar in terms of contouring performance, but the SMC controller produced more than double the mean contour error of the other two controllers.

4.5.2 End-Effector Payload Simulations

A simulated load of 1.5 [kg] was added at the end-effector for both the serial and parallel manipulator to simulate a change in the dynamics of the original systems. The results of the simulation with an added payload for the serial manipulator can be seen in Table 4.56 and Figure 4.84. The parallel manipulator results are shown in Table 4.57 and Figure 4.85.

Table 4.56: Magnitude of Contour Tracking Performance for Motion with a Payload of Serial Manipulator

	Mean (m)	S.D. (m)	$\%(\varepsilon \leq 2(10^{-4}) [m])$
PD	0.0002082	$7.323(10^{-5})$	37.14%
SMC	0.0002190	0.0001068	22.82%
PD-SMC	0.0001654	$6.049(10^{-5})$	43.79%

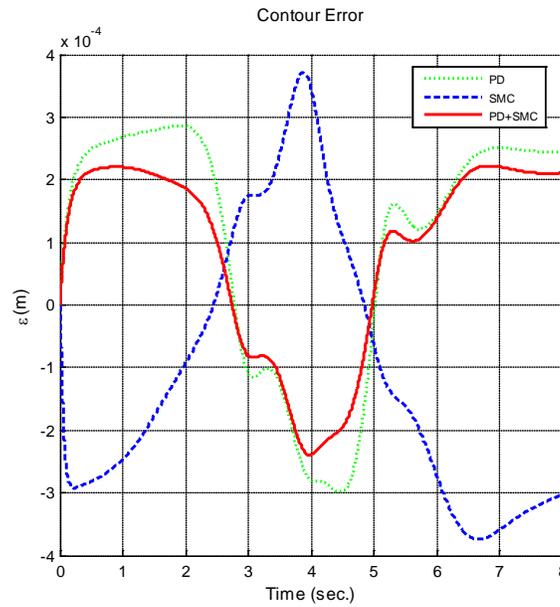


Figure 4.84: Contour Error for Motion with a Payload of the Serial Manipulator

Table 4.57: Magnitude of Contour Tracking Performance for Motion with a Payload of Parallel Manipulator

	Mean (m)	S.D. (m)	$\%(\varepsilon \leq 1.5(10^{-4}) [m])$
PD	0.0001058	$8.436(10^{-5})$	77.14%
SMC	0.0001246	$8.842(10^{-5})$	84.38%
PD-SMC	$9.104(10^{-5})$	$6.813(10^{-5})$	80.88%

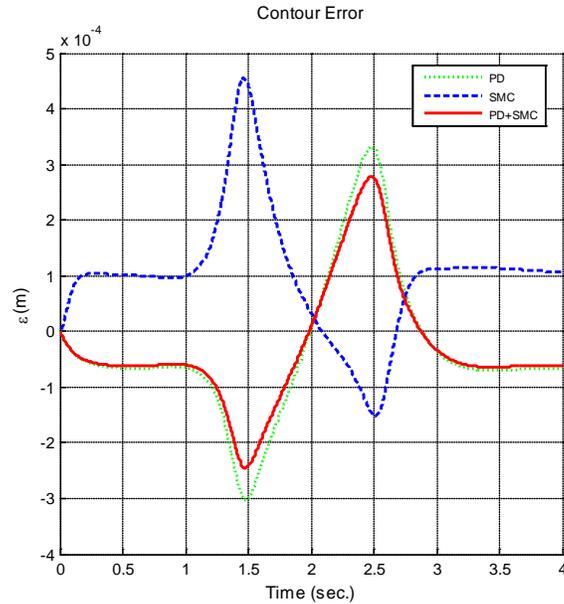


Figure 4.85: Contour Error for Motion with a Payload of the Parallel Manipulator

Similar to the high-speed motion results, the SMC controller produced the largest increase in mean contour error magnitudes and standard deviations. Although the mean contour error for the serial manipulator for the PD controller increased by 60%, and 75% for the PD-SMC controller, the PD-SMC retained the best contour tracking performance of the three manipulators. For the parallel manipulator, the PD-SMC mean contour tracking error magnitude was the only one that remained the same magnitude as for the original simulation. The payload results indicate that the PD-SMC controller is robust to changes in the system dynamics.

4.5.3 High Control Gains

For the first set of simulations, the PD gains were increased by 50% while the SMC control parameters remained fixed to their previous values. The performance results of this simulation can be seen in Table 4.58 and Figure 4.86 for the serial manipulator and Table 4.59 and Figure 4.87 for the parallel manipulator. The PD gains were then set to their previous values and the SMC parameters were increased by 50%. These results are shown in Table 4.60 and

Figure 4.88 for the serial manipulator and Table 4.61 and Figure 4.89 for the parallel manipulator.

Table 4.58: Magnitude of Contour Tracking Performance for Motion with High PD Gains for Serial Manipulator

	Mean (m)	S.D. (m)	$\%(\varepsilon \leq 1.5(10^{-4}) [m])$
PD	$8.749(10^{-5})$	$3.726(10^{-5})$	89.81%
SMC	$9.830(10^{-5})$	$4.918(10^{-5})$	83.12%
PD-SMC	$6.932(10^{-5})$	$2.334(10^{-5})$	100%

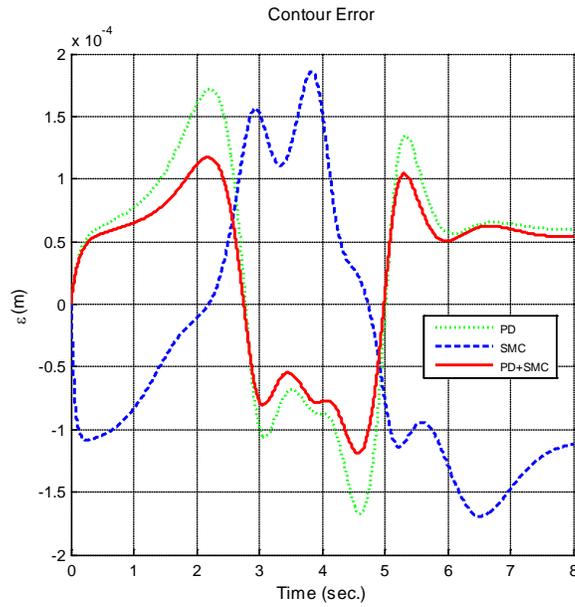


Figure 4.86: Contour Error for Motion with a High PD Gains for Serial Manipulator

Table 4.59: Magnitude of Contour Tracking Performance for Motion with High PD Gains for Parallel Manipulator

	Mean (m)	S.D. (m)	$\%(\varepsilon \leq 1.5(10^{-4}) [m])$
PD	$3.098(10^{-5})$	$3.789(10^{-5})$	98.23%
SMC	$3.760(10^{-5})$	$5.054(10^{-5})$	93.45%
PD-SMC	$2.677(10^{-5})$	$3.432(10^{-5})$	100%

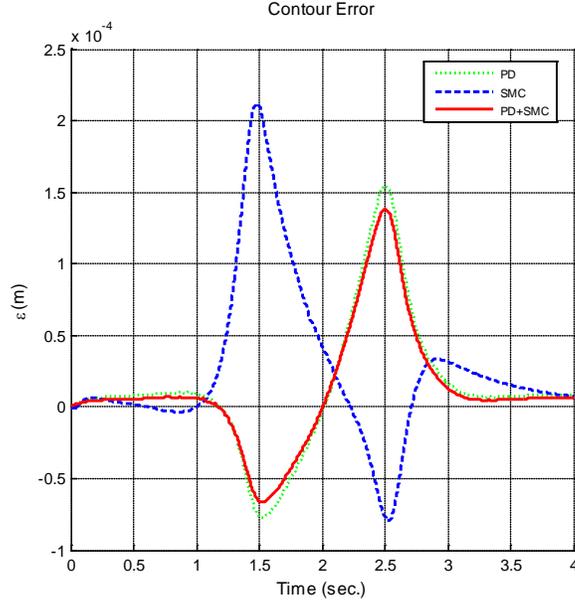


Figure 4.87: Contour Error for Motion with a High PD Gains for Parallel Manipulator

Since there are no PD gains for the SMC controller, the performance results for the SMC controller are identical to those of the initial simulation for this movement. The means of the contour error magnitudes, as well as the standard deviations for these errors, for the PD controller in both the serial and parallel manipulator decreased by about 33%. For the PD-SMC controller the mean contour errors decreased more than 25% for both the serial and parallel manipulator. The standard deviation of the contour errors for the serial manipulator decreased by 20%. For the parallel manipulator, the standard deviation decreased by 30%. These results verify that increasing the PD gains generally improves performance for the PD and PD-SMC controllers, as previously indicated in Section 4.2.3.

Table 4.60: Magnitude of Contour Tracking Performance for Motion with High SMC Gains for Serial Manipulator

	Mean (m)	S.D. (m)	$\%(\epsilon \leq 2(10^{-4}) [m])$
PD	0.0001313	$5.590(10^{-5})$	84.64%
SMC	$4.317(10^{-5})$	$2.167(10^{-5})$	100%
PD-SMC	$7.201(10^{-5})$	$1.817(10^{-5})$	100%

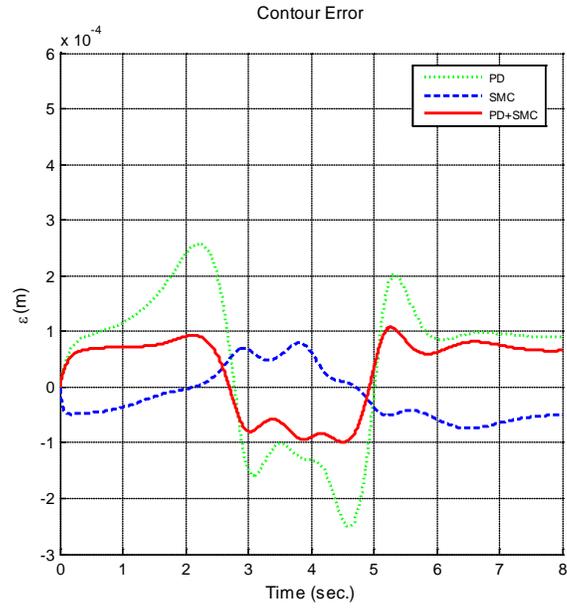


Figure 4.88: Contour Error for Motion with a High SMC Gains for Serial Manipulator

Table 4.61: Magnitude of Contour Tracking Performance for Motion with High SMC Gains for Parallel Manipulator

	Mean (m)	S.D. (m)	$\%(\varepsilon \leq 1.5(10^{-4}) [m])$
PD	$4.637(10^{-5})$	$5.698(10^{-5})$	92.03%
SMC	$1.701(10^{-5})$	$2.323(10^{-5})$	100%
PD-SMC	$2.953(10^{-5})$	$4.150(10^{-5})$	96.54%

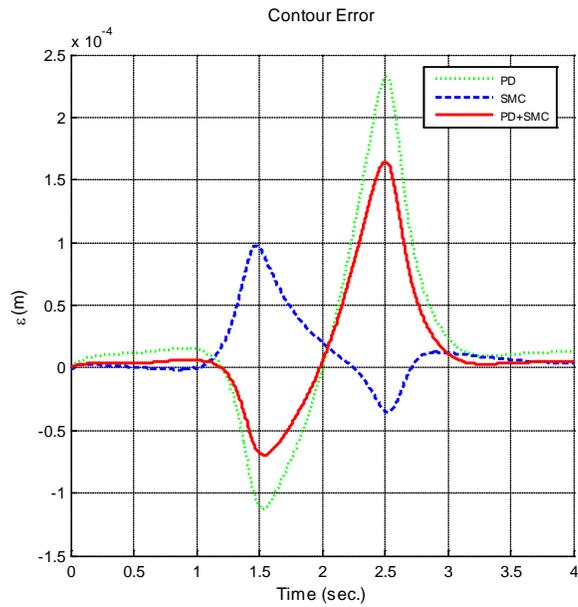


Figure 4.89: Contour Error for Motion with a High SMC Gains for Parallel Manipulator

An increase in the SMC control parameters yielded smaller mean contour errors for the PD-SMC controller. For the serial manipulator the mean contour error magnitude was reduced by 24% and for the parallel manipulator it was reduced by 20%. For the SMC controller, the mean contour error magnitudes were reduced by more than 50% in each case. The PD-SMC controller, however, retained the best contour tracking performance for the serial manipulator. Although the SMC controller produced the best results for the parallel manipulator, the PD-SMC controller also demonstrated improved performance and the difference in performances remain only marginal. This large change in performance, however, shows the sensitivity of the SMC controller to changes in the gains. This further verifies the good contour tracking performance and superior robustness of the PD-SMC controller to changes and uncertainties in the dynamics of the system.

4.6 REMARKS

The simulation results indicate a general improvement in tracking and contouring performance of the proposed hybrid PD-SMC controller over standard PD and SMC control schemes. It should be noted that for the SMC controller, theoretically perfect tracking can be achieved if the dynamics of the system are perfectly known (Slotine & Li, 1991). However, with the addition of simulated estimations of the system dynamics, as well as the addition of a boundary layer for trajectory smoothing, the PD-SMC achieved overall better contour tracking performance.

For the serial manipulator, the PD-SMC controller produced an average of 27% lower mean contour error magnitudes up to 44% lower than the PD controller. The PD-SMC controller also yielded an average of 21% smaller mean contour error magnitudes than the SMC controller.

In terms of the standard deviation of these errors, the PD-SMC controller produced roughly 15% to 50% smaller values than the standard PD controller and 10% to 50% smaller values than the SMC controller. For the parallel manipulator the PD-SMC controller produced up to 20% smaller mean contour errors than the PD controller and up to 14% smaller mean values than the SMC controller. Comparing standard deviations of contour error magnitudes, the PD-SMC controller produced an average of 20% smaller standard deviation values than both the PD and SMC controllers. A comparison of the mean contour errors is shown in Figure 4.90 and Figure 4.91 for the serial and parallel manipulators, respectively.

It was also demonstrated that the maximum torques required for the serial manipulator were generally lower for the PD and PD-SMC controllers than the SMC controller, with the exception of axis 1. For the parallel manipulator, the torque requirements were about equal for each controller. However, it should be noted that for the PD-SMC controller there was higher control activity demonstrated by a fluctuating input torque. This controller activity was smoothed by using a higher sampling rate and can be smoothed further by further increasing the sampling rate. Because the input torques, as well as the maximum torque values, were similar for each controller, it indicates that the power requirements for the proposed PD-SMC controller are similar to those standard PD and SMC controllers.

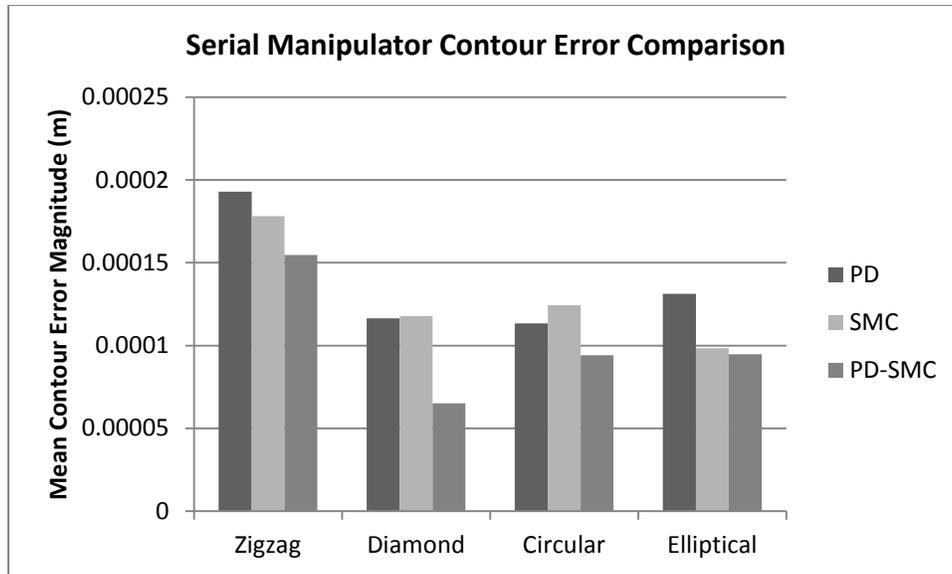


Figure 4.90: Serial Manipulator Mean Contour Error Magnitude Performance Comparison

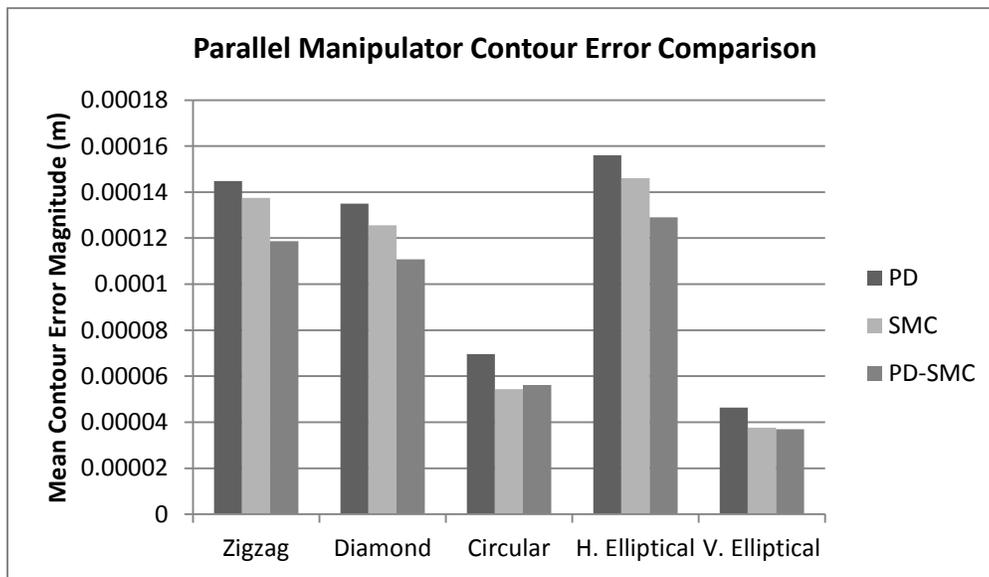


Figure 4.91: Parallel Manipulator Mean Contour Error Magnitude Performance Comparison

The results from the high-speed motion, end-effector payload, and higher gain simulations demonstrated the robustness of the proposed PD-SMC controller. The SMC controller was shown to be highly sensitive to the changes in the dynamic parameters of the system while the PD-SMC controller maintained good contour tracking results despite the changes to the initial dynamics for which it was originally tuned.

Chapter 5 : CONCLUSIONS AND FUTURE WORK

5.1 GENERAL REVIEW

In this thesis, a new control law was proposed for the control of multi-DOF robotic manipulators. This was done by the hybridization of two control systems: PD and SMC. By satisfying the complementary and compatibility principles of the hybridization of control systems, the weaknesses of each of the existing control systems are complemented by the inherent strength of the other. As a result, the proposed hybrid PD-SMC law provides an intuitive, model-free alternative to the SMC law as well as improved tracking performance relative to the standard PD controller with the introduction of a nonlinear switching term. The stability of the proposed PD-SMC law was demonstrated with the Lyapunov method.

Simulations using linear and nonlinear contours were conducted using the new control law for a planar serial 3R manipulator and a planar parallel 5R manipulator. The tracking and contour performances of the PD-SMC law were compared to the performances of the PD and SMC control laws. By introducing simulated estimations of the system dynamics, uncertainties from friction and other unmodeled dynamics were added to the SMC simulations. The PD-SMC law demonstrated significant improvement in tracking performance relative to standard PD control while producing comparable performance to the SMC law. It was shown that in the cases where the SMC law produced smaller tracking errors, it did not necessarily result in better contouring performance. The PD-SMC law demonstrated consistent improvement over the other controllers in terms of contour tracking performance.

With the addition of higher-speed movements and end-effector payloads, it was further shown that the PD-SMC maintained good contour tracking results. The robustness of the

proposed law was especially evident when compared to the SMC law, which showed significantly degraded performance quality when introduced to unmodeled dynamics. The PD-SMC law consistently demonstrated marked improvements in both contouring performance and robustness over the other two control laws.

5.2 MAIN CONTRIBUTIONS

The following contributions are made in this thesis:

- An alternative to standard PD control with the inclusion of a nonlinear switching term.
- A model-free alternative to standard SMC law with control parameters that have intuitive meanings for simple tuning.
- Easily implemented, robust control law with improved tracking and contouring performances for robotic manipulators for linear and nonlinear contours.

5.3 FUTURE WORK

The work presented in this thesis shows that a hybridized control law can provide intuitive and model-free control of robotic manipulators with improved tracking and contouring performance over PD control and improved robustness over SMC control. Further work should be carried out to demonstrate the effectiveness for different configurations of robots, such as spatial manipulators. Tracking of arbitrary contours should also be explored. Next, experimental results should be obtained to validate the practical application of the proposed control law. Finally, as position domain controllers increase in popularity, the proposed law may be

transformed into the position domain where using master axis motion as an independent reference instead of time can further improve contouring performance.

LIST OF APPENDICES

SERIAL 3R MANIPULATOR KINEMATICS AND DYNAMICS

Forward Kinematics

$$x_{ee} = L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) + L_3 \cos(q_1 + q_2 + q_3) \quad (1)$$

$$y_{ee} = L_1 \sin(q_1) + L_2 \sin(q_1 + q_2) + L_3 \sin(q_1 + q_2 + q_3) \quad (2)$$

$$q_{ee} = q_1 + q_2 + q_3 \quad (3)$$

Inverse Kinematics

Position:

$$x_3 = x_{ee} - L_3 \cos(q_{ee}) \quad (4)$$

$$y_3 = y_{ee} - L_3 \sin(q_{ee}) \quad (5)$$

Then,

$$q_1 = \text{atan2}\left(\frac{-y_3}{\sqrt{x_3^2 + y_3^2}}, \frac{-x_3}{\sqrt{x_3^2 + y_3^2}}\right) \pm \cos^{-1}\left(\frac{-(x_3^2 + y_3^2 + L_1^2 - L_2^2)}{2L_1\sqrt{x_3^2 + y_3^2}}\right) \quad (6)$$

$$q_2 = \text{atan2}\left(\frac{y_3 - L_1 \sin(q_1)}{L_2}, \frac{x_3 - L_1 \cos(q_1)}{L_2}\right) - q_1 \quad (7)$$

$$q_3 = q_{ee} - q_1 - q_2 \quad (8)$$

Velocity:

$$\dot{q}_1 = \frac{\cos(q_1 + q_2)}{L_1 \sin(q_2)} \dot{x}_{ee} + \frac{\sin(q_1 + q_2)}{L_1 \sin(q_2)} \dot{y}_{ee} + \frac{L_3 \sin(q_3)}{L_1 \sin(q_2)} \dot{q}_{ee} \quad (9)$$

$$\begin{aligned} \dot{q}_2 = & -\frac{L_2 \cos(q_1 + q_2) + L_1 \cos(q_1)}{L_1 L_2 \sin(q_2)} \dot{x}_{ee} - \frac{L_2 \sin(q_1 + q_2) + L_1 \sin(q_1)}{L_1 L_2 \sin(q_2)} \dot{y}_{ee} \\ & - \frac{L_3 \sin(q_2 + q_3) + L_3 \cos(q_3)}{L_1 L_2 \sin(q_2)} \dot{q}_{ee} \end{aligned} \quad (10)$$

$$\dot{q}_3 = \frac{\cos(q_1)}{L_2 \sin(q_2)} \dot{x}_{ee} + \frac{\sin(q_1)}{L_2 \sin(q_2)} \dot{y}_{ee} + \left(\frac{L_3 \sin(q_2 + q_3)}{L_2 \sin(q_2)} + 1 \right) \dot{q}_{ee} \quad (11)$$

Dynamics

Inertia Matrix:

$$\begin{aligned} M_{11} = & m_1 r_1^2 + m_2 (L_1^2 + r_2^2 + 2L_1 L_2 \cos(q_2)) \\ & + m_3 (L_1^2 + L_2^2 + r_3^2 + 2L_1 L_2 \cos(q_2) + 2L_2 r_3 \cos(q_3) \\ & + 2L_1 r_3 \cos(q_2 + q_3)) + I_1 + I_2 + I_3 \end{aligned} \quad (12)$$

$$\begin{aligned} M_{12} = M_{21} = & m_2 (r_2^2 + L_1 r_2 \cos(q_2)) \\ & + m_3 (L_2^2 + r_3^2 + 2L_1 L_2 \cos(q_2) + 2L_2 r_3 \cos(q_3) + 2L_1 r_3 \cos(q_2 + q_3)) \\ & + I_2 + I_3 \end{aligned} \quad (13)$$

$$M_{13} = M_{31} = m_3 (r_3^2 + 2L_3 \cos(q_3) + 2L_1 r_3 \cos(q_2 + q_3)) + I_3 \quad (14)$$

$$M_{22} = m_2 r_2^2 + m_3 (L_2^2 + r_3^2 + 2L_2 r_3 \cos(q_3)) + I_2 + I_3 \quad (15)$$

$$M_{23} = M_{32} = m_3 (L_2^2 + r_3^2 + 2L_2 r_3 \cos(q_3)) + I_3 \quad (16)$$

$$M_{33} = m_3 r_3^2 + I_3 \quad (17)$$

Centrifugal-Coriolis Terms:

$$C(q, \dot{q}) = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix} \quad (18)$$

Let:

$$\begin{Bmatrix} h_1 \\ h_2 \\ h_3 \end{Bmatrix} = \begin{bmatrix} -(m_2 L_1 r_2 + m_3 L_1 L_2) \sin(q_2) \\ -m_3 L_1 r_3 \sin(q_2 + q_3) \\ -m_3 L_2 r_3 \sin(q_3) \end{bmatrix} \quad (19)$$

Then,

$$C_1 = (h_1 + h_2)(2\dot{q}_1 + \dot{q}_2)\dot{q}_2 + (h_2 + h_3)(2\dot{q}_1 + 2\dot{q}_2 + \dot{q}_3)\dot{q}_3 \quad (20)$$

$$C_2 = -(h_1 + h_2)\dot{q}_1^2 + h_3(2\dot{q}_1 + 2\dot{q}_2 + \dot{q}_3)\dot{q}_3 \quad (21)$$

$$C_3 = -(h_2 + h_3)\dot{q}_1^2 - h_3(2\dot{q}_1 + \dot{q}_2)\dot{q}_2 \quad (22)$$

Gravity Vector:

$$G_{11} = g[m_1 r_1 \cos(q_1) + m_2(L_1 \cos(q_1) + r_2 \cos(q_1 + q_2)) + m_3(L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) + r_3 \cos(q_1 + q_2 + q_3))] \quad (23)$$

$$G_{12} = g[m_2 r_2 \cos(q_1 + q_2) + m_3(L_2 \cos(q_1 + q_2) + r_3 \cos(q_1 + q_2 + q_3))] \quad (24)$$

$$G_{13} = g m_3 r_3 \cos(q_1 + q_2 + q_3) \quad (25)$$

PARALLEL 5R MANIPULATOR KINEMATICS AND DYNAMICS

Forward Kinematics

Let:

$$\lambda = L_2 \cos(q_2) - L_1 \cos(q_1) + L_5 \quad (26)$$

$$\mu = L_2 \sin(q_2) - L_1 \sin(q_1) \quad (27)$$

$$X = 2L_4 \lambda \quad (28)$$

$$Y = 2L_4 \mu \quad (29)$$

$$Z = L_3^2 - L_4^2 - \lambda^2 - \mu^2 \quad (30)$$

Then,

$$q_4 = \text{atan2}\left(\pm\sqrt{X^2 + Y^2 - Z^2}, Z\right) + \text{atan2}(Y, X) - q_2 \quad (31)$$

$$q_3 = \text{atan2}(\mu + L_4 \sin(q_2 + q_4), \lambda + L_4 \cos(q_2 + q_4)) - q_1 \quad (32)$$

$$x_{ee} = L_1 \cos(q_1) + L_3 \cos(q_1 + q_3) \quad (33)$$

$$y_{ee} = L_1 \sin(q_1) + L_3 \sin(q_1 + q_3) \quad (34)$$

Inverse Kinematics

Position:

Let:

$$P = \sqrt{(-L_1^2 + 2L_1L_3 - L_3^2 + x_{ee}^2 + y_{ee}^2)(L_1^2 + 2L_1L_3 + L_3^2 - x_{ee}^2 - y_{ee}^2)} \quad (35)$$

$$Q = \left[(L_2^2 + 2L_2L_4 + L_4^2 - L_5^2 + 2L_5x_{ee} - x_{ee}^2 - y_{ee}^2)(-L_2^2 + 2L_2L_4 - L_4^2 + L_5^2 - 2L_5x_{ee} + x_{ee}^2 + y_{ee}^2) \right]^{\frac{1}{2}} \quad (36)$$

Then,

$$x_3 = \frac{L_1^2 - L_3^2 + x_{ee}^2 + y_{ee}^2 - \left(\frac{L_1^2 y_{ee}^2 - L_3^2 y_{ee}^2 + x_{ee}^2 y_{ee}^2 + y_{ee}^4 \pm y_{ee} x_{ee} P}{x_{ee}^2 + y_{ee}^2} \right)}{2x_{ee}} \quad (37)$$

$$y_3 = \frac{y_{ee}(L_1^2 - L_3^2 + x_{ee}^2 + y_{ee}^2) \pm x_{ee} P}{2(x_{ee}^2 + y_{ee}^2)} \quad (38)$$

x_4

$$= \frac{1}{2}(x_{ee} - L_5)^{-1} \left[L_2^2 - L_4^2 - L_5^2 + x_{ee}^2 + y_{ee}^2 - \left(\frac{L_2^2 y_{ee}^2 - L_5^2 y_{ee}^2 + x_{ee}^2 y_{ee}^2 - y_{ee}^4 - 2L_5 x_{ee} y_{ee}^2 \pm y_{ee}(x_{ee} - L_5)Q}{L_5^2 - 2L_5 x_{ee} + x_{ee}^2 + y_{ee}^2} \right) \right] \quad (39)$$

$$y_4 = \frac{y_{ee}(L_2^2 - L_4^2 + L_5^2 + x_{ee}^2 + y_{ee}^2 - 2L_5 x_{ee}) \pm (x_{ee} - L_5)Q}{2(L_5^2 - 2L_5 x_{ee} + x_{ee}^2 + y_{ee}^2)} \quad (40)$$

$$q_1 = \text{atan2}(y_3, x_3) \quad (41)$$

$$q_2 = \text{atan2}(y_4, x_4 - L_5) \quad (42)$$

Use forward kinematics to calculate q_3 and q_4 .

Velocity:

$$\dot{q}_1 = \frac{\dot{x}_{ee} \cos(q_1 + q_3) + \dot{y}_{ee} \sin(q_1 + q_3)}{L_1(\cos(q_1) \sin(q_1 + q_3) - \sin(q_1) \cos(q_1 + q_3))} \quad (43)$$

$$\dot{q}_2 = \frac{\dot{x}_{ee} \cos(q_2 + q_4) + \dot{y}_{ee} \sin(q_2 + q_4)}{L_2(\cos(q_2) \sin(q_2 + q_4) - \sin(q_2) \cos(q_2 + q_4))} \quad (44)$$

$$\dot{q}_3 = -\frac{\dot{x}_{ee}(L_1 \cos(q_1) + L_3 \cos(q_1 + q_3)) + \dot{y}_{ee}(L_1 \sin(q_1) + L_3 \sin(q_1 + q_3))}{L_1 L_3(\cos(q_1) \sin(q_1 + q_3) - \sin(q_1) \cos(q_1 + q_3))} \quad (45)$$

$$\dot{q}_4 = -\frac{\dot{x}_{ee}(L_2 \cos(q_2) + L_4 \cos(q_2 + q_4)) + \dot{y}_{ee}(L_2 \sin(q_2) + L_4 \sin(q_2 + q_4))}{L_2 L_4(\cos(q_2) \sin(q_2 + q_4) - \sin(q_2) \cos(q_2 + q_4))} \quad (46)$$

Dynamics

Inertia Matrix:

From (Ghorbel, 1995), we have:

$$\psi_{q'}(q') = \begin{bmatrix} \psi_{q'}(1,1) & \psi_{q'}(1,2) & \psi_{q'}(1,3) & \psi_{q'}(1,4) \\ \psi_{q'}(2,1) & \psi_{q'}(2,2) & \psi_{q'}(2,3) & \psi_{q'}(2,4) \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (47)$$

Where:

$$\psi_{q'}(1,1) = -L_1 \sin(q_1) - L_3 \sin(q_1 + q_3) \quad (48)$$

$$\psi_{q'}(1,2) = L_2 \sin(q_2) + L_4 \sin(q_2 + q_4) \quad (49)$$

$$\psi_{q'}(1,3) = -L_3 \sin(q_1 + q_3) \quad (50)$$

$$\psi_{q'}(1,4) = L_4 \sin(q_2 + q_4) \quad (51)$$

$$\psi_{q'}(2,1) = L_1 \cos(q_1) + L_3 \cos(q_1 + q_3) \quad (52)$$

$$\psi_{q'}(2,2) = -L_2 \cos(q_2) - L_4 \cos(q_2 + q_4) \quad (53)$$

$$\psi_{q'}(2,3) = L_3 \cos(q_1 + q_3) \quad (54)$$

$$\psi_{q'}(2,4) = -L_4 \cos(q_2 + q_4) \quad (55)$$

The parameter $\dot{\psi}_{q'}(q', \dot{q}')$ is obtained by differentiating Eq. (47) with respect to time.

$$\rho(q') = \psi_{q'}^{-1}(q') \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (56)$$

$$\dot{\rho}(q', \dot{q}') = -\psi_{q'}^{-1}(q') \dot{\psi}_{q'}(q', \dot{q}') \rho(q') \quad (57)$$

Now,

$$M'(q') = \begin{bmatrix} M_{11} & 0 & M_{13} & 0 \\ 0 & M_{22} & 0 & M_{24} \\ M_{31} & 0 & M_{33} & 0 \\ 0 & M_{42} & 0 & M_{44} \end{bmatrix} \quad (58)$$

Where:

$$M_{11} = m_1 r_1^2 + m_3 (L_1^2 + r_3^2 + 2L_1 r_3 \cos(q_3)) + I_1 + I_3 \quad (59)$$

$$M_{13} = M_{31} = m_3 (r_3^2 + L_1 r_3 \cos(q_3)) + I_3 \quad (60)$$

$$M_{22} = m_2 r_2^2 + m_4 (L_2^2 + r_4^2 + 2L_2 r_4 \cos(q_4)) + I_2 + I_4 \quad (61)$$

$$M_{24} = M_{42} = m_4 (r_4^2 + L_2 r_4 \cos(q_4)) + I_4 \quad (62)$$

$$M_{33} = m_3 r_3^2 + I_3 \quad (63)$$

$$M_{44} = m_4 r_4^2 + I_4 \quad (64)$$

Then,

$$M(q') = \rho(q')^T M'(q') \rho(q') \quad (65)$$

Centrifugal-Coriolis Terms:

Let:

$$\begin{Bmatrix} h_1 \\ h_2 \end{Bmatrix} = \begin{bmatrix} -m_3 L_1 r_3 \sin(q_3) \\ -m_4 L_2 r_4 \sin(q_4) \end{bmatrix} \quad (66)$$

Then,

$$C'(q', \dot{q}') = \begin{bmatrix} h_1 \dot{q}_3 & 0 & h_1 (\dot{q}_1 + \dot{q}_3) & 0 \\ 0 & h_2 \dot{q}_4 & 0 & h_2 (\dot{q}_2 + \dot{q}_4) \\ -h_1 \dot{q}_1 & 0 & 0 & 0 \\ 0 & -h_2 \dot{q}_2 & 0 & 0 \end{bmatrix} \quad (67)$$

$$C(q', \dot{q}') = \rho(q')^T C'(q', \dot{q}') \rho(q') + \rho(q')^T M'(q') \dot{\rho}(q', \dot{q}') \quad (68)$$

Gravity Vector:

$$G'(q') = g \begin{bmatrix} (m_1 r_1 + m_3 L_1) \cos(q_1) + m_3 r_3 \cos(q_1 + q_3) \\ (m_2 r_2 + m_4 L_2) \cos(q_2) + m_4 r_4 \cos(q_2 + q_4) \\ m_3 r_3 \cos(q_1 + q_3) \\ m_4 r_4 \cos(q_2 + q_4) \end{bmatrix} \quad (69)$$

$$G(q') = \rho(q')^T G'(q') \quad (70)$$

BIBLIOGRAPHY

- Alici, G. (2002). An inverse position analysis of five-bar planar parallel manipulators. *Robotica*, 21(2), 195-201.
- Allamehzadeh, H., & Cheung, J. Y. (2002). Chattering-Free Sliding Mode Fuzzy Control with Continuous Inherent Boundary Layer. *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems* (pp. 1393-1398). IEEE.
- Angeles, J. (2007). *Fundamentals of Robotic Mechanical Systems: Theory Methods, and Algorithms (3rd Edition)*. New York: Springer.
- Antsaklis, P. J., & Nerode, A. (1998). Hybrid Control Systems: An Introductory Discussion to the Special Issue. *IEEE Transactions on Automatic Control*, 43(4), 457-460.
- Arimoto, S., & Miyazaki, F. (1984). Stability and robustness of PID feedback control for robot manipulators of sensory capability. *Robotics Research: The First International Symposium*, (pp. 783-799).
- Arimoto, S., Kawamura, S., & Miyazaki, F. (1984). Bettering operation of robots by learning. *Journal of Robotic Systems*, 1(2), 123-140.
- Armstrong, B., & Wade, B. A. (2000). Nonlinear PID Control with Partial State Knowledge: Damping without Derivatives. *The International Journal of Robotics Research*, 19(8), 715-731.
- Årzén, K. E. (1999). A simple event-based PID controller. *Proc. 14th IFAC World Congress*, 18, pp. 423-428.
- Åström, K. J. (1983). Theory and Applications of Adaptive Control - A Survey. *Automatica*, 19(5), 471-486.

- Åström, K. J., & Hägglund, T. (2001). The future of PID control. *Control Engineering Practice*, 9(11), 1163-1175.
- Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.
- Bristow, D. A., Tharayil, M., & Alleyne, A. G. (2006). A survey of iterative learning control. *Control Systems, IEEE*, 26(3), 96-114.
- Chen, Q., Chen, H., Wang, Y., & Woo, P. (2000). Global Stability Analysis for Some Trajectory-Tracking Control Schemes of Robotic Manipulators. *Proceedings of the American Control Conference*, (pp. 3343-3347). Chicago.
- Cheng, M. Y., & Lee, C. C. (2005). On Real-time Contour Error Estimation for Contour Following Tasks. *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Machines*, (pp. 1047-1052).
- Craig, J. J. (1988). *Adaptive Control of Mechanical Manipulators*. Addison-Wesley.
- Craig, J. J. (2005). *Introduction to Robotics, Mechanics, and Control* (Third ed.). Pearson Prentice Hall.
- Dessaint, L. A., Saad, M., Hébert, B., & Al-Haddad, K. (1992). An Adaptive Controller for a Direct-Drive Scara Robot. *IEEE Transactions on Industrial Electronics*, 39(2), 105-111.
- Ghorbel, F. (1995). Modeling and PD Control of Closed-Chain Mechanical Systems. *Proc. of the 34th IEEE Conference on Decision and Control*, (pp. 540-542). New Orleans.
- Heemels, W., Sandee, J. H., & Van Den Bosch, P. (2008). Analysis of event-driven controllers for linear systems. *International Journal of Control*, 81(4), 571-590.
- Hsu, C.-F., Chiu, C.-J., & Tsai, J.-Z. (2011). Auto-tuning PID controller design using a sliding-mode approach for DC servomotors. *International Journal of Intelligent Computing and Cybernetics*, 4(1), 93-110.

- International Federation of Robotics. (2013). *World Robotics 2013 Industrial Robots*. IFR Statistical Department.
- Kawamura, S., Miyazaki, F., & Arimoto, S. (1988). Is a local linear PD feedback control law effective for trajectory tracking of robot motion? *IEEE International Conference on Robotics and Automation*, (pp. 1335-1340).
- Kelly, R. (1997). PD Control with Desired Gravity Compensation of Robotic Manipulators: A Review. *The International Journal of Robotics Research*, 16(5), 660-672.
- Koren, Y., & Lo, C. C. (1991). Variable-gain cross-coupling controller for contouring. *CIRP Annals-Manufacturing Technology*, 40(1), 371-374.
- Kuc, T. Y., & Han, W. G. (2000). An adaptive PID learning control of robot manipulators. *Automatica*, 36(5), 717-725.
- Li, Y., Ng, K. C., Murray-Smith, D. J., Gray, G. J., & Sharman, K. C. (1996). Genetic algorithm automated approach to the design of sliding mode control systems. *International Journal of Control*, 63(4), 721-739.
- Ouyang, P. (2005). *Hybrid Intelligent Machine Systems: Design, Modeling and Control*. PhD Thesis, University of Saskatchewan, Department of Mechanical Engineering.
- Pervozvanski, A. A., & Freidovich, L. B. (1999). Robust Stabilization of Robotic Manipulators by PID Controllers. *Dynamics and Control*, 9(3), 203-222.
- Qu, Z. (1994). Global Stability of Trajectory Tracking of Robot Under PD Control. *Dynamics and Control*, 4(1), 59-71.
- Qu, Z., & Dorsey, J. (1991). Robust PID control of robots. *International Journal of Robotics and Automation*, 6(4), 228-235.

- Rugh, W. (1987). Design of nonlinear PID controllers. *American Institute of Chemical Engineers Journal*, 33(10), 1738-1742.
- Sage, H. G., De Mathelin, M. F., & Ostertag, E. (1999). Robust Control of Robot Manipulators: A Survey. *International Journal of Control*, 72(16), 1498-1522.
- Sastry, S., & Bodson, M. (1989). *Adaptive Control: Stability, Convergence, and Robustness*. Prentice-Hall, Inc.
- Seraji, H. (1998). Nonlinear and Adaptive Control of Force and Compliance in Manipulators. *The International Journal of Robotics Research*, 17(5), 467-484.
- Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2009). *Robotics: Modelling, Planning and Control*. London: Springer.
- Slotine, J.-J. E., & Li, W. (1988). Adaptive Manipulator Control: A Case Study. *IEEE Transactions on Automatic Control*, 33(11), 995-1003.
- Slotine, J.-J. E., & Li, W. (1991). *Applied Nonlinear Control*. New Jersey: Prentice-Hall, Inc.
- Stiver, J. A., & Antsaklis, P. J. (1992). Modeling and Analysis of Hybrid Control Systems. *Proceedings of the 31st IEEE Conference on Decision and Control*. 4, pp. 3748-3751. IEEE.
- Takegaki, M., & Arimoto, S. (1981). A new feedback method for dynamics control of manipulators. *ASME, Transactions, Journal of Dynamic Systems, Measurement and Control*, 103, 119-125.
- Ting, K.-L. (1989). Mobility Criteria of Single-Loop N-Bar Linkages. *Journal of Mechanisms, Transmissions and Automation in Design*, 111(4), 504-507.
- Tomei, P. (1991). Adaptive PD Controller for Robot Manipulators. *IEEE Transactions on Robotics and Automation*, 7(4), 565-570.

- Wang, J., Mounier, H., Cela, A., & Niculescu, S. I. (2011). Event driven intelligent PID controllers with applications to motion control. *18th IFAC World Congress*. Milan.
- White, D. A., & Sofge, D. A. (1992). *Handbook of Intelligent Control*. Van Nostrand Reinhold.
- Xu, Y., Hollerbach, J. M., & Ma, D. (1995). A nonlinear PD controller for force and contact transient control. *Control Systems, IEEE*, 15(1), 15-21.
- 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.