



## City Research Online

### City, University of London Institutional Repository

---

**Citation:** Osman, J.H.S. (1991). Decentralized and hierarchical control of robot manipulators. (Unpublished Doctoral thesis, City University London)

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

---

**Permanent repository link:** <https://openaccess.city.ac.uk/id/eprint/7758/>

**Link to published version:**

**Copyright:** City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

**Reuse:** Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

# **DECENTRALIZED AND HIERARCHICAL CONTROL OF ROBOT MANIPULATORS**

020174016

**BY**

**JOHARI HALIM SHAH OSMAN,  
B.Sc. (Physics), M.Sc. (Elec. Eng.)**

**A THESIS SUBMITTED FOR THE DEGREE OF  
DOCTOR OF PHILOSOPHY IN ENGINEERING**

**CONTROL ENGINEERING CENTRE  
SCHOOL OF ELECTRICAL ENGINEERING  
AND APPLIED PHYSICS  
CITY UNIVERSITY  
LONDON**

**OCTOBER 1991.**

To:

Izzaz

&

Nadira

# TABLE OF CONTENTS

	Page
TABLE OF CONTENTS	2
LIST OF TABLES	6
LIST OF FIGURES	7
ACKNOWLEDGEMENTS	20
DECLARATION	21
PUBLICATIONS	22
ABSTRACT	23
LIST OF SYMBOLS	24
 CHAPTER 1 INTRODUCTION	 40
1.1 Robotic System	40
1.2 Manipulator Link Dynamic Model	42
1.3 Manipulator Control Strategies	46
1.4 Research Objectives	54
1.5 Structure and Layout of Thesis	56
 CHAPTER 2 MODELLING OF ROBOT MANIPULATOR	 58
2.1 Introduction	58
2.2 State Space representation of Manipulator Link Dynamics	59
2.3 Actuator Dynamics	61
2.3.1 Third Order Actuator Model With $\theta, \dot{\theta}, i_a$ As State Variables	66
2.3.2 Third Order Actuator Model With $\theta, \dot{\theta}, \ddot{\theta}$ As State Variables	67
2.4 Integrated Dynamic Model Of Robot Manipulator	70
2.4.1 Integrated Model Based On $\theta, \dot{\theta}, i_a$ As State Variables	71
2.4.2 Integrated Model Based On $\theta, \dot{\theta}, \ddot{\theta}$ As State Variables	74
2.4.3 Differences Between The Two Integrated Models	77



2.5	Dynamic Model Of A Three dof Revolute Robot Manipulator	78
2.5.1	Dynamic Equation Of The Mechanical Linkage	78
2.5.2	Derivative Of The Mechanical Link Torque	85
2.5.3	Model Of The DC Motors	87
2.5.4	Integrated Model Of the Three dof Robot Manipulator	89
2.6	Conclusion	91
 <b>CHAPTER 3 ROBOT MANIPULATOR AS A LARGE SCALE UNCERTAIN SYSTEM</b>		 <b>93</b>
3.1	Introduction	93
3.2	Decomposition Of Robot Manipulator	94
3.2.1	Decomposition Based On Integrated Model With Joint Angle, Velocity And Current As State Variables	96
3.2.2	Decomposition Based On Integrated Model With Joint Angle, Velocity And Acceleration As State Variables	98
3.3	Robot Manipulator As An Uncertain Interconnected System	102
3.4	Application To A Three dof Robot Manipulator	108
3.5	Conclusion	112
 <b>CHAPTER 4 DECENTRALIZED TRACKING CONTROLLER DESIGN FOR ROBOTIC MANIPULATORS</b>		 <b>114</b>
4.1	Introduction	114
4.2	Deterministic Control Of Uncertain Systems	115
4.2.1	Deterministic Control Approach Of Leitmann [1981], And Corless And Leitmann [1981]	116
4.2.2	Application Of A Deterministic Approach To Robot Manipulator Control – An Overview	118
4.3	Problem Formulation	119
4.4	Decentralized Nonlinear Tracking Controller Design – Local Approach	124
4.5	Simulation And Results	134
4.5.1	Trajectory Generation	135
4.5.2	Simulation Using Independent Joint Linear Control	139

4.5.3 Simulation Using Decentralized Nonlinear Local Control Method	143
4.5.3.1 Effect Of Load Variation	151
4.5.3.2 Effect Of Varying The Value $\epsilon_i$	159
4.5.3.3 Effect Of Varying The Weighting Matrix $Q_i$	162
4.6 Conclusion	167
 <b>CHAPTER 5 DECENTRALIZED GLOBAL TRACKING CONTROLLER DESIGN FOR ROBOT MANIPULATOR</b>	 168
5.1 Introduction	168
5.2 Problem Formulation	169
5.3 Decentralized Nonlinear Tracking Controller Design – Global Approach	171
5.4 Simulation And Results	180
5.4.1 Comparison Between Local And Global Approach	183
5.4.2 Effect Of Load Variation	189
5.4.3 Effect Of Varying The Value $\epsilon_i$	196
5.4.4 Effect Of Varying The Weighting Matrix $Q_i$	203
5.4.5 Comparison Between The Decentralized And A Centralized Control Method	207
5.5 Conclusion	215
 <b>CHAPTER 6 HIERARCHICAL CONTROL CONCEPT IN ROBOTIC SYSTEM</b>	 216
6.1 Introduction	216
6.2 Problem Formulation	217
6.3 Hierarchical Control Structure – Method 1	222
6.3.1 Controller Design At The Lowest Level – Linear Feedback Control	222
6.3.2 Controller Design At The Upper Level – Deterministic Approach	224
6.4 Hierarchical Control Structure – Method 2	227
6.4.1 Controller Design At The Lowest Level – Deterministic Approach	227

6.4.2 Controller Design At The Upper Level – Deterministic Approach	231
6.4.3 Stability Of The Overall System	232
6.5 Simulation And Results	238
6.5.1 Implementation Of The First Hierarchical Control Method	238
6.5.2 Implementation Of The Second Hierarchical Control Method	242
6.5.3 Effect Of Load Variation	247
6.5.4 Effect Of Structural Perturbation	256
6.5.5 Effect Of Varying The Sampling Interval At The Upper Level	260
6.5.6 Comparison Between The Proposed Control Methods	272
6.6 conclusion	275
<b>CHAPTER 7 CONCLUSION AND SUGGESTIONS</b>	<b>276</b>
7.1 Conclusion	276
7.2 Suggestions For Future Research	279
7.3 Concluding Remarks	280
<b>APPENDIX A DENAVIT-HARTENBERG (D-H) NOTATION AND TRANSFORMATION MATRIX</b>	<b>281</b>
<b>APPENDIX B EULER-LAGRANGE FORMULATION OF MANIPULATOR LINKS DYNAMICS</b>	<b>285</b>
<b>APPENDIX C ROBOT MANIPULATOR COMPLETE MODEL – A SURVEY</b>	<b>290</b>
<b>APPENDIX D DETAILED EXPRESSIONS OF THE INTEGRATED DYNAMIC MODEL OF THE THREE DOF REVOLUTE ROBOT MANIPULATOR</b>	<b>294</b>
<b>REFERENCES</b>	<b>297</b>

# LIST OF TABLES

	Page
<u>CHAPTER 2</u>	
Table 2-1 : Parameters Of The Three dof Robot Manipulator Linkage	80
Table 2-2 : Parameters Of The Actuators	81

<u>CHAPTER 5</u>	
Table 5-1 : Different Sets Of $\epsilon_i$ And The Corresponding $\eta_G$	198
Table 5-2 : Different Values Of $Q_i$ And The Corresponding $\eta_G$	204

# LIST OF FIGURES

	Page
<u>CHAPTER 1</u>	
Figure 1.1 : Robotic System Components	41
Figure 1.2 : $N$ dof Robot Manipulator	45
<u>CHAPTER 2</u>	
Figure 2.1 : Schematic Diagram Of Armature Controlled Permanent Magnet DC Motor	63
Figure 2.2 : Diagram Of The $i$ th Manipulator Link Actuated By DC Motor With Gear Train	64
Figure 2.3 : Three dof Revolute Robot Manipulator	79
<u>CHAPTER 4</u>	
Figure 4.1 : Decentralized Control Of Robot Manipulator – Local Approach	126
Figure 4.2 : Desired Joint Position Profile	137
Figure 4.3 : Desired Joint Velocity Profile	137
Figure 4.4 : Desired Joint Acceleration Profile	138
Figure 4.5 : Joint Tracking Error Without Load Using Linear Control Method	142

Figure 4.6	: Joint Tracking Error For 20 Kg. Load Using Linear Controller	142
Figure 4.7	: Position Tracking Responses For 10 Kg. Load Using Decentralized Local Control Method	148
Figure 4.8	: Velocity Tracking Responses For 10 Kg. Load Using Decentralized Local Control Method	148
Figure 4.9	: Acceleration Tracking Responses For 10 Kg. Load Using Decentralized Local Control Method	149
Figure 4.10	: Position Tracking Error For 10 Kg. Load	149
Figure 4.11	: Joint Control Inputs For 10 Kg. Load Using Decentralized Local Control Method	150
Figure 4.12a	: Joint 1 Tracking Response Under Different Load Using Decentralized Local Control Method	152
Figure 4.12b	: Joint 1 Tracking Error Under Different Load Using Decentralized Local Control Method	152
Figure 4.13a	: Joint 2 Tracking Response Under Different Load Using Decentralized Local Control Method	153
Figure 4.13b	: Joint 2 Tracking Error Under Different Load Using Decentralized Local Control Method	153
Figure 4.14a	: Joint 3 Tracking Response Under Different Load Using Decentralized Local Control Method	154
Figure 4.14b	: Joint 3 Tracking Error Under Different Load Using Decentralized Local Control Method	154

Figure 4.15a	: Joint 1 Velocity Response Under Different Load Using Decentralized Local Control Method	155
Figure 4.15b	: Joint 1 Acceleration Error Under Different Load Using Decentralized Local Control Method	155
Figure 4.16a	: Joint 2 Velocity Response Under Different Load Using Decentralized Local Control Method	156
Figure 4.16b	: Joint 2 Acceleration Error Under Different Load Using Decentralized Local Control Method	156
Figure 4.17a	: Joint 3 Velocity Response Under Different Load Using Decentralized Local Control Method	157
Figure 4.17b	: Joint 3 Acceleration Error Under Different Load Using Decentralized Local Control Method	157
Figure 4.18	: Control Inputs For 0 Kg. Load Using Decentralized Local Control Method	158
Figure 4.19	: Control Inputs For 20 Kg. Load Using Decentralized Local Control Method	158
Figure 4.20	: Joint 1 Tracking Error For 10 Kg. Load For Different Values Of $\epsilon_i$	160
Figure 4.21	: Joint 2 Tracking Error For 10 Kg. Load For Different Values Of $\epsilon_i$	160
Figure 4.22	: Joint 3 Tracking Error For 10 Kg. Load For Different Values Of $\epsilon_i$	161

Figure 4.23	: Joint Control Inputs For 10 Kg. Load For $\epsilon_1 = 0.25$ , $\epsilon_2 = 0.2$ , $\epsilon_3 = 0.54$	161
Figure 4.24	: Joint Tracking Errors For 10 Kg. Load For $\epsilon_1 = 0.2$ , $\epsilon_2 = 0.15$ , $\epsilon_3 = 0.5$	163
Figure 4.25	: Joint Control Inputs For 10 Kg. Load For $\epsilon_1 = 0.2$ , $\epsilon_2 = 0.15$ , $\epsilon_3 = 0.5$	163
Figure 4.26	: Joint 1 Tracking Error For 10 Kg. Load For Different $Q_i$ Matrix	165
Figure 4.27	: Joint 2 Tracking Error For 10 Kg. Load For Different $Q_i$ Matrix	165
Figure 4.28	: Joint 3 Tracking Error For 10 Kg. Load For Different $Q_i$ Matrix	166
Figure 4.29	: Joint Control Inputs For 10 Kg. Load For $Q_i = 1.1 I_3$	166

## CHAPTER 5

Figure 5.1	: Decentralized Control Of Robot Manipulator – Global Approach	173
Figure 5.2a	: Joint 1 Tracking Response Under Decentralized Local And Global Control Methods	184
Figure 5.2b	: Joint 1 Tracking Error Under Decentralized Local And Global Control Methods	184
Figure 5.3a	: Joint 2 Tracking Response Under Decentralized Local And Global Control Methods	185



Figure 5.3b	: Joint 2 Tracking Error Under Decentralized Local And Global Control Methods	185
Figure 5.4a	: Joint 3 Tracking Response Under Decentralized Local And Global Control Methods	186
Figure 5.4b	: Joint 3 Tracking Error Under Decentralized Local And Global Control Methods	186
Figure 5.5	: Velocity Tracking Responses For 10 Kg. Load Using Decentralized Global Controller	187
Figure 5.6	: Acceleration Tracking Responses For 10 Kg. Load Using Decentralized Global Control Method	187
Figure 5.7	: Joint Control Inputs For 10 Kg. Load Using Decentralized Global Control Method	188
Figure 5.8a	: Joint 1 Tracking Response Under Different Load Using Decentralized Global Control Method	190
Figure 5.8b	: Joint 1 Tracking Error Under Different Load Using Decentralized Global Control Method	190
Figure 5.9a	: Joint 2 Tracking Response Under Different Load Using Decentralized Global Control Method	191
Figure 5.9b	: Joint 2 Tracking Error Under Different Load Using Decentralized Global Control Method	191
Figure 5.10a	: Joint 3 Tracking Response Under Different Load Using Decentralized Global Control Method	192

Figure 5.10b	: Joint 3 Tracking Error Under Different Load Using Decentralized Global Control Method	192
Figure 5.11a	: Joint 1 Velocity Response Under Different Load Using Decentralized Global Control Method	193
Figure 5.11b	: Joint 1 Acceleration Error Under Different Load Using Decentralized Global Control Method	193
Figure 5.12a	: Joint 2 Velocity Response Under Different Load Using Decentralized Global Control Method	194
Figure 5.12b	: Joint 2 Acceleration Error Under Different Load Using Decentralized Global Control Method	194
Figure 5.13a	: Joint 3 Velocity Response Under Different Load Using Decentralized Global Control Method	195
Figure 5.13b	: Joint 3 Acceleration Error Under Different Load Using Decentralized Global Control Method	195
Figure 5.14	: Control Inputs For 0 Kg. Load Using Decentralized Global Control Method	197
Figure 5.15	: Control Inputs For 20 Kg. Load Using Decentralized Global Control Method	197
Figure 5.16	: Joint 1 Tracking Error For 10 Kg. Load For Different Values Of $\epsilon_i$	200
Figure 5.17	: Joint 2 Tracking Error For 10 Kg. Load For Different Values Of $\epsilon_i$	200

Figure 5.18	: Joint 3 Tracking Error For 10 Kg. Load For Different Values Of $\epsilon_i$	201
Figure 5.19	: Joint Control Inputs For 10 Kg. Load For $\epsilon_1 = 0.6$ , $\epsilon_2 = 0.3$ , $\epsilon_3 = 0.7$	201
Figure 5.20	: Joint Tracking Errors For 10 Kg. Load For $\epsilon_1 = 0.4$ , $\epsilon_2 = 0.15$ , $\epsilon_3 = 0.5$	202
Figure 5.21	: Joint Control Inputs For 10 Kg. Load For $\epsilon_1 = 0.4$ , $\epsilon_2 = 0.3$ , $\epsilon_3 = 0.5$	202
Figure 5.22	: Joint 1 Tracking Error For 10 Kg. Load For Different $Q_i$ Matrix	205
Figure 5.23	: Joint 2 Tracking Error For 10 Kg. Load For Different $Q_i$ Matrix	205
Figure 5.24	: Joint 3 Tracking Error For 10 Kg. Load For Different $Q_i$ Matrix	206
Figure 5.25	: Joint Control Inputs For 10 Kg. Load For $Q_i = 1.1 I_3$	206
Figure 5.26	: Joint 1 Tracking Error Using Different Decentralized And Centralized Control Methods	210
Figure 5.27	: Joint 2 Tracking Error Using Different Decentralized And Centralized Control Methods	210
Figure 5.28	: Joint 3 Tracking Error Using Different Decentralized And Centralized Control Methods	211
Figure 5.29	: Joint 1 Tracking Error Using Different Decentralized And Centralized Control Methods	213

Figure 5.30	: Joint 2 Tracking Error Using Different Decentralized And Centralized Control Methods	213
Figure 5.31	: Joint 3 Tracking Error Using Different Decentralized And Centralized Control Methods	214

## CHAPTER 6

Figure 6.1	: Decentralized Linear Controller For The $i$ th Decoupled Subsystem At The Lowest Level Using Method 1	223
Figure 6.2	: Two – Level Hierarchical Control Of Robot Manipulator – Method 1	226
Figure 6.3	: Decentralized Nonlinear Controller For The $i$ th Decoupled Subsystem At The Lowest Level Using Method 2	229
Figure 6.4	: Two – Level Hierarchical Control Of Robot Manipulator – Method 2	237
Figure 6.5	: Position Tracking Responses For 10 Kg. Load Using Hierarchical Control – Method 1	240
Figure 6.6	: Position Tracking Error For 10 Kg. Load Using Hierarchical Control – Method 1	240
Figure 6.7	: Velocity Tracking Responses For 10 Kg. Load Using Hierarchical Control – Method 1	241
Figure 6.8	: Acceleration Tracking Responses For 10 Kg. Load Using Hierarchical Control – Method 1	241

Figure 6.9	: Joint Control Inputs For 10 Kg. Load Using Hierarchical Control – Method 1	242
Figure 6.10	: Position Tracking Responses For 10 Kg. Load Using Hierarchical Control – Method 2	244
Figure 6.11	: Position Tracking Error For 10 Kg. Load Using Hierarchical Control – Method 2	245
Figure 6.12	: Velocity Tracking Responses For 10 Kg. Load Using Hierarchical Control – Method 2	245
Figure 6.13	: Acceleration Tracking Responses For 10 Kg. Load Using Hierarchical Control – Method 2	246
Figure 6.14	: Joint Control Inputs For 10 Kg. Load Using Hierarchical Control – Method 2	246
Figure 6.15a	: Joint 1 Tracking Response Under Different Load Using Hierarchical Control – Method 1	248
Figure 6.15b	: Joint 1 Tracking Error Under Different Load Using Hierarchical Control – Method 1	248
Figure 6.16a	: Joint 2 Tracking Response Under Different Load Using Hierarchical Control – Method 1	249
Figure 6.16b	: Joint 2 Tracking Error Under Different Load Using Hierarchical Control – Method 1	249
Figure 6.17a	: Joint 3 Tracking Response Under Different Load Using Hierarchical Control – Method 1	250

Figure 6.17b	: Joint 3 Tracking Error Under Different Load Using Hierarchical Control – Method 1	250
Figure 6.18	: Control Inputs For 0 Kg. Load Using Hierarchical Control – Method 1	251
Figure 6.19	: Control Inputs For 20 Kg. Load Using Hierarchical Control – Method 1	251
Figure 6.20a	: Joint 1 Tracking Response Under Different Load Using Hierarchical Control – Method 2	252
Figure 6.20b	: Joint 1 Tracking Error Under Different Load Using Hierarchical Control – Method 2	252
Figure 6.21a	: Joint 2 Tracking Response Under Different Load Using Hierarchical Control – Method 2	253
Figure 6.21b	: Joint 2 Tracking Error Under Different Load Using Hierarchical Control – Method 2	253
Figure 6.22a	: Joint 3 Tracking Response Under Different Load Using Hierarchical Control – Method 2	254
Figure 6.22b	: Joint 3 Tracking Error Under Different Load Using Hierarchical Control – Method 2	254
Figure 6.23	: Control Inputs For 0 Kg. Load Using Hierarchical Control – Method 2	255
Figure 6.24	: Control Inputs For 20 Kg. Load Using Hierarchical Control – Method 2	255

Figure 6.25	: Joint 1 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 1	257
Figure 6.26	: Joint 2 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 1	257
Figure 6.27	: Joint 3 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 1	258
Figure 6.28	: Joint 1 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 2	258
Figure 6.29	: Joint 2 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 2	259
Figure 6.30	: Joint 3 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 2	259
Figure 6.31	: Joint 1 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 1	261
Figure 6.32	: Joint 2 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 1	262
Figure 6.33	: Joint 3 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 1	262
Figure 6.34a	: Velocity Tracking Response When The Sampling Rate Of The Coordinator Is At 0.05s – Method 1	263
Figure 6.34b	: Acceleration Tracking Response When The Sampling Rate Of The Coordinator Is At 0.05s – Method 1	263

Figure 6.35a	: Velocity Tracking Response When The Sampling Rate Of The Coordinator Is At 0.1s – Method 1	264
Figure 6.35b	: Acceleration Tracking Response When The Sampling Rate Of The Coordinator Is At 0.1s – Method 1	264
Figure 6.36	: Joint Control Inputs When The Sampling Rate Of The Coordinator Is At 0.05s – Method 1	265
Figure 6.37	: Joint Control Inputs When The Sampling Rate Of The Coordinator Is At 0.1s – Method 1	265
Figure 6.38	: Joint 1 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 2	266
Figure 6.39	: Joint 2 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 2	266
Figure 6.40	: Joint 3 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 2	267
Figure 6.41	: Joint 1 Velocity Response Under Different Rates Of Computation Of The Coordinator – Method 2	267
Figure 6.42	: Joint 2 Velocity Response Under Different Rates Of Computation Of The Coordinator – Method 2	268
Figure 6.43	: Joint 3 Velocity Response Under Different Rates Of Computation Of The Coordinator – Method 2	268
Figure 6.44	: Joint 1 Acceleration Response Under Different Rates Of Computation Of The Coordinator – Method 2	269



Figure 6.45	: Joint 2 Acceleration Response Under Different Rates Of Computation Of The Coordinator – Method 2	269
Figure 6.46	: Joint 3 Acceleration Response Under Different Rates Of Computation Of The Coordinator – Method 2	270
Figure 6.47	: Joint Control Inputs When The Sampling Rate Of The Coordinator Is At 0.05s – Method 2	270
Figure 6.48	: Joint Control Inputs When The Sampling Rate Of The Coordinator Is At 0.1s – Method 2	271
Figure 6.49	: Joint 1 Tracking Errors Using Decentralized, Hierarchical, And Centralized Control Methods	274
Figure 6.50	: Joint 2 Tracking Errors Using Decentralized, Hierarchical, And Centralized Control Methods	274
Figure 6.51	: Joint 3 Tracking Errors Using Decentralized, Hierarchical, And Centralized Control Methods	275

## APPENDIX A

Figure A.1	: Denavit-Hartenberg Frame Assignment [Spong and Vidyasagar, 1989]	284
------------	---	-----

# ACKNOWLEDGEMENTS

I would like to express my thanks and gratitude to my supervisor, Professor P. D. Roberts, for his invaluable guidance, patience, and encouragement throughout this project.

I would like to thank the Universiti Teknologi of Malaysia (U.T.M.) for supporting this research in the form of a scholarship. There are many people at U.T.M., amongst them Associate Professor Dr. Mohammad Nor Bin Salleh, to whom I owe thanks for helping me secure the scholarship.

My thanks are also due to Mrs. J. Rivellini, Iman Kartowisastro, and other staff at the Control Engineering Centre for their help.

I would also like to thank my parents for their enduring patience, understanding, and for taking care of my children, Izzaz and Nadira.

Finally, I would like to express my utmost appreciation to my wife for her constant moral support, encouragement and help, and to my daughter Yazmin, thanks for being wonderful.

# DECLARATION

The author grants powers of discretion to the University Librarian to allow this thesis to be copied in whole or in part without further reference to him. This permission covers only single copies made for study purposes, subject to normal conditions of acknowledgement.

## PUBLICATION

The following paper, based on the work described in this thesis, has been published :

1. Osman, J. H. S., and Roberts, P. D. [1991] : A Class Of Decentralized Tracking Controller For Robot Manipulators. *Proc. Inst. Mech. Engrs., Part 1, J. of Systems and Control Engineering*, pp. 141 - 150.

# ABSTRACT

This thesis deals with the decentralized and hierarchical control of a class of robot manipulators, where the robot manipulator is treated as a large scale uncertain system. The work is divided into three parts.

The first part is concerned with the development of an integrated mathematical model of the robot manipulator. The model of the system considered comprises the mechanical part of the robot manipulator, the actuators, as well as the gear trains. The formulation results in nonlinear time varying state equations, which represent a more realistic model of the robotic system. A procedure to decompose and reduce the integrated model of the robot manipulator into a set of interconnected subsystems with bounded uncertainties description is then presented.

In the second part of the research, two decentralized control approaches based on a deterministic approach are outlined. The first method uses only the local states as the feedback information. It is shown that the robot manipulator utilizing the proposed controller is practically stable and tracks a reference trajectory if a given sufficient condition is satisfied. In the second approach, the controller is designed based on the local states as well as the states of the neighbouring subsystems as the feedback information. It is shown that the controller will force the nonlinear uncertain robot manipulator to track a desired trajectory to within a small uniform ultimate boundedness set.

In the final part of the study, two hierarchical control concepts for robot manipulator are proposed. The controllers are formulated based on a deterministic approach. It is shown that the hierarchical control strategies are capable of withstanding the expected variations and uncertainties and will render the robot manipulator to track a prescribed trajectory satisfactorily.

In synthesizing the proposed controllers, it is assumed that the upper bounds on the nonlinearities, couplings and uncertainties present in the system are available. The proposed methods are simple and robust to parameter variations and uncertainties present in the system. The performance of the proposed control algorithms are evaluated by means of computer simulations. The proposed control laws are applied to a three degree of freedom revolute robot manipulator actuated by DC motors. Several case studies have been considered, and the simulation results are presented and discussed.

In this thesis, the term practical stability means bounded stability in the sense of Lyapunov.

# LIST OF SYMBOLS

<u>SYMBOL</u>	<u>DESCRIPTION</u>
1. Upper Case :	
$A(*,*,*)$	$3N \times 3N$ system matrix for the integrated robot manipulator model
$A_A(*,*,*)$	$3N \times 3N$ system matrix for the integrated robot manipulator model using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$A_{Aij}(*,*,*)$	$ij$ th submatrix of $A_A(*,*,*)$
$A_i(*,*,*)$	system matrix for the $i$ th subsystem
$A_{ij}(*,*,*)$	interconnection matrix between the $i$ th and $j$ th subsystems
$A_A$	$3N \times 3N$ system matrix for the augmented dynamic equation of the actuators using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$A_{Ai}$	$3 \times 3$ system matrix for the $i$ th actuators using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$A_B(*,*,*)$	$3N \times 3N$ system matrix for the integrated robot manipulator model using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$A_{Bij}(*,*,*)$	$ij$ th submatrix of $A_B(*,*,*)$
$A_B$	$3N \times 3N$ system matrix for the augmented dynamic equation of the actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables

$A_{Bi}$	3x3 system matrix for the $i$ th actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$A_i$	nominal system matrix for the $i$ th subsystem
$A_p(*,*,*)$	$2N \times 2N$ system matrix for the mechanical linkage dynamics expressed in state state variable form
$A_{p1}(*,*,*)$	$N \times N$ matrix component of the matrix $A_p(*,*,*)$
$A_{p2}(*,*,*)$	$N \times N$ matrix component of the matrix $A_p(*,*,*)$
$\tilde{A}_i$	closed loop system matrix for the $i$ th subsystem
$\Delta A_i(*,*)$	matrix representing the uncertainties in the system matrix for the $i$ th subsystem
$A_{i-1}^i$	Denavit-Hartenberg transformation matrix relating the $i$ th coordinate frame to the $i-1$ th coordinate frame
$B(*,*,*)$	$3N \times N$ input matrix for the integrated robot manipulator model
$B_A$	$3N \times N$ input matrix for the augmented dynamic equation of the actuators using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$B_{Ai}$	$3 \times 1$ input matrix for the $i$ th actuators using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$B_B(*,*,*)$	$3N \times N$ input matrix for the integrated robot manipulator model using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$B_{Bij}(*,*,*)$	$ij$ th submatrix of $B_B(*,*,*)$

$B_B$	$3N \times N$ input matrix for the augmented dynamic equation of the actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$B_{B_i}$	$3 \times 1$ input matrix for the $i$ th actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$B_i(*,*,*)$	input matrix for the $i$ th subsystem
$B_i$	nominal input matrix for the $i$ th subsystem
$\Delta B_i(*,*)$	matrix representing the uncertainties in the input matrix for the $i$ th subsystem
$B_p(*,*,*)$	$2N \times N$ input matrix for the mechanical linkage dynamics expressed in state variable form
$B_p^\dagger(*,*,*)$	Penrose-Pseudoinverse of $B_p(*,*,*)$
$B_{v_i}$	viscous friction coefficient for the $i$ th motor (Nm/rad/s)
$\mathfrak{B}(\eta)$	close ball with radius $\eta$
$\tilde{C}(*,*,*)$	$N \times N$ coefficient matrix related to the acceleration vector in the derivative of the manipulator link dynamic equation
$\tilde{C}_{ij}$	$ij$ th element of the matrix $\tilde{C}(*,*,*)$
$D(*,*,*)$	$N \times 1$ vector of Coriolis and centrifugal forces
$\bar{D}(*,*)$	$N \times \sum_{i=1}^N i$ matrix related to $D(*,*,*)$ vector
$\bar{D}_{ij}$	$ij$ th element of the matrix $\bar{D}(*,*)$
$\dot{\bar{D}}(*,*)$	derivative of the matrix $\bar{D}(*,*)$ with respect to time



$\hat{D}(*,*,*)$	$N \times N$ matrix related to $D(*,*,*)$ vector
$\tilde{D}(*,*,*)$	$N \times N$ coefficient matrix related to the velocity vector in the derivative of the manipulator link dynamic equation
$\tilde{D}_{ij}$	$ij$ th element of the matrix $\tilde{D}(*,*,*)$
$E_i(*,*)$	a continuous function related to $\Delta B_i(*,*)$
$\mathcal{E}_i$	back emf for the $i$ th actuator
$F_A$	$3N \times N$ load distribution matrix for the augmented dynamic equation of the actuators using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$F_{Ai}$	$3 \times 1$ load distribution matrix for the $i$ th actuators using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$F_B$	$3N \times N$ load distribution matrix for the augmented dynamic equation of the actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$F_{Bi}$	$3 \times 1$ load distribution matrix for the $i$ th actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$G(*,*)$	$N \times 1$ vector of gravitational forces
$G_i$	$i$ th component of the matrix $G(*,*)$
$\dot{G}(*,*)$	derivative of $G(*,*)$ with respect to time
$\hat{G}(*,*)$	$N \times N$ matrix related to the vector of gravitational forces $G(*,*)$
$G_{ij}(*,*)$	$1 \times 3$ vector of continuous functions related to the interconnection matrix $A_{ij}(*,*)$

$H_i(*,*)$	1x3 vector of continuous functions related to $\Delta A_i(*,*)$
$I_i$	an $i \times i$ identity matrix
$\mathbb{I}_{11}^i, \mathbb{I}_{22}^i, \mathbb{I}_{33}^i$	moment of inertia at the centre of gravity for the $i$ th link with respect to x-, y-, and z-axis, respectively
$\mathfrak{J}$	set denoting the number of joint (degree of freedom) of the manipulator
$J_{mi}$	moment of inertia for the $i$ th motor ( Kgm <sup>2</sup> )
$\mathbb{J}_i$	pseudo inertia matrix of the $i$ th link
$K_i$	linear feedback gain matrix for the $i$ th subsystem
$\mathcal{K}(*,*)$	total kinetic energy of the system
$L(*,*)$	Lagrange function
$L_i$	armature inductance for the $i$ th motor (H)
$L_{fi}$	field inductance for the $i$ th motor (H)
$\mathcal{L}$	Lyapunov function
$\dot{\mathcal{L}}$	derivative of the Lyapunov function $\mathcal{L}$
$M(*,*)$	$N \times N$ inertia matrix of the manipulator linkage
$M_{ij}$	$ij$ th elements of matrix $M(*,*)$
$\dot{M}(*,*)$	derivative of the inertia matrix $M(*,*)$

$N_i$	inverse of the gear ratio (gear ratio= $1/N_i$ ) for the $i$ th joint
$N$	number of degree of freedom/joints of the robot manipulator
$O_{ij}$	an $ixj$ null matrix
$P_i$	solution of matrix Lyapunov equation for the $i$ th subsystem
$P$	$3N \times 3N$ matrix with $P_i$ , $i=1,2,\dots,N$ as the diagonal submatrices and null off-diagonal submatrices
$\tilde{P}$	solution of the matrix Lyapunov equation for the overall system
$\mathcal{P}(\cdot)$	total potential energy of the system
$Q$	positive definite symmetric weighting matrix,
$R_i$	armature resistance for the $i$ th motor ( $\Omega$ )
$\mathcal{R}$	uncertainty bounding set for $\Delta a_{ij}^i(\cdot, \cdot)$ for all $i$ and $j$
$\mathcal{R}^N$	$N$ -dimensional real space
$\mathcal{Y}$	uncertainty bounding set for $\Delta b_i^i(\cdot, \cdot)$ for all $i$ .
$T(\cdot)$	$N \times 1$ vector of driving forces/torques applied by the actuators at the drive points on each link of the manipulator
$T_i(\cdot)$	$i$ th component of $T(\cdot) \equiv$ driving forces/torques applied to the $i$ th joint by the $i$ th actuator
$T_{Li}(\cdot)$	load torque on the motor shaft at the primary side of the $i$ th motor (Nm)

$\dot{T}(\cdot)$	derivative of the manipulator torque $T(\cdot)$ with respect to time
$\dot{T}_i(\cdot)$	$i$ th component of $\dot{T}(\cdot)$
$\text{Tr}$	trace operator
$\mathbb{T}_L$	$N \times N$ test matrix for decentralized local control law
$\mathbb{T}_G$	$N \times N$ test matrix for decentralized global control law
$\mathbb{T}_{H2}$	$N \times N$ test matrix for the second hierarchical control strategy
$U$	$N \times 1$ control input vector for an $N$ degree of freedom robot manipulator
$U_i(\cdot)$	control input for the $i$ th actuator/joint $\equiv i$ th element of $U(\cdot)$
$U_{Li}(\cdot)$	decentralized local controller for the $i$ th subsystem
$U_{Gi}(\cdot)$	decentralized global controller for the $i$ th subsystem
$U_i^L(\cdot)$	control for the $i$ th subsystem generated at the lower level using two-level hierarchical control structure
$U_i^{L1}(\cdot)$	control for the $i$ th subsystem generated at the lower level using the first hierarchical control method
$U_i^{L2}(\cdot)$	control for the $i$ th subsystem generated at the lower level using the second hierarchical control method
$U_i^U(\cdot)$	control for the $i$ th subsystem generated at the upper level using two-level hierarchical control structure

$U_i^{U1}(*)$	control for the $i$ th subsystem generated at the upper level using the first hierarchical control method
$U_i^{U2}(*)$	control for the $i$ th subsystem generated at the upper level using the second hierarchical control method
$\tilde{U}_i(*,*,*)$	linear control component for the $i$ th subsystem
$U_{ij}$	first partial derivative of the transformation matrix with respect to joint variable
$U_{ijk}$	second partial derivative of the transformation matrix with respect to joint variable
$V(*)$	$\sum_{i=1}^N i \times N$ velocity matrix related to $D(*,*,*)$ vector
$\dot{V}(*)$	derivative of the matrix $V(*)$ with respect to time
$\mathbb{V}$	$N \times N$ diagonal matrix
$\mathbb{V}_{H2}$	$N \times N$ diagonal matrix obtained using the second hierarchical control strategy
$\mathcal{V}$	uncertainty bounding set for $\Delta a_{ij}^{ij}(*,*)$ for all $i$ and $j$
$W_B$	$3N \times N$ rate of load distribution matrix for the augmented dynamic equation of the actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$W_{Bi}$	$3 \times 1$ rate of load distribution matrix for the $i$ th actuators using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$X_p(*)$	$2N \times 1$ state vector for the manipulator linkage

$X_A(*)$	$3N \times 1$ state vector for the integrated robot manipulator model using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$X_{Ai}(*)$	$3 \times 1$ state vector for the $i$ th actuator using $\theta_i$ , $\dot{\theta}_i$ , and $i_{ai}$ as the state variables
$X_B(*)$	$3N \times 1$ state vector for the integrated robot manipulator model using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$X_{Bi}(*)$	$3 \times 1$ state vector for the $i$ th actuator using $\theta_i$ , $\dot{\theta}_i$ , and $\ddot{\theta}_i$ as the state variables
$X_d(*)$	$3N \times 1$ desired state trajectory vector for the integrated robot manipulator system
$X_{di}(*)$	$3 \times 1$ desired state trajectory vector for the $i$ th subsystem
$X_i(*)$	$3 \times 1$ state vector for the $i$ th subsystem
$X_{io}$	$3 \times 1$ initial state vector for the $i$ th subsystem
$\mathfrak{E}(\kappa)$	Lyapunov ellipsoid with respect to $\kappa$
$Z(*)$	$3N \times 1$ error state vector between the actual and the desired states of the overall system
$Z_i(*)$	$3 \times 1$ error state vector between the actual and the desired states for the $i$ th subsystem
$Z_A$	$2N \times 3N$ transformation matrix relating $X_p(t)$ to $X_A(t)$
$Z_B$	$N \times 3N$ transformation matrix relating $\ddot{\theta}(t)$ to $X_B(t)$
$Z_{B1}$	$N \times 3N$ transformation matrix relating $\dot{\theta}(t)$ to $X_B(t)$

$Z_{B2}$	$N \times 3N$ transformation matrix relating $\theta(t)$ to $X_B(t)$
$Z_C$	$N \times 3N$ transformation matrix relating $\ddot{\theta}(t)$ to $\dot{X}_A(t)$
$\tilde{Z}$	$3N \times 1$ vector whose elements are $\ Z_i\ $ , $i=1, 2, \dots, N$
$(*)^T$	transpose of $(*)$
$(*)^c$	complement of $(*)$
$\ (*)\ $	Euclidean norm of $(*)$

## 2. Lower Case :

$a_i$	common normal distance along $x_i$ -axis from the intersection of the $x_i$ and $z_{i-1}$ -axes to the origin of the $i$ th coordinate frame
$a_{Bij}$	$ij$ th element of the matrix $A_B$
$a_{ij}$	$ij$ th element of the integrated system matrix $A(*,*,*)$
$a_{ij}^i(*,*)$	$ij$ th element of the system matrix $A_i(*,*)$ for the $i$ th subsystem
$a_{ij}^{ij}(*,*)$	$ij$ th element of the $ij$ th interconnection matrix $A_{ij}(*,*)$
$\underline{a}_{ij}^i, \bar{a}_{ij}^i$	minimum and maximum bound of $a_{ij}^i(*,*)$ , respectively
$a_{ij}^i$	$ij$ th element of the nominal system matrix $A_i$ for the $i$ th subsystem
$\Delta a_{ij}^i(*,*)$	$ij$ th element of the uncertain matrix $\Delta A_i(*,*)$ for the $i$ th subsystem

$b_{Bi}$	element of the input matrix $B_{Bi}$ for the $i$ th actuator
$b_{ij}$	$ij$ th element of the integrated input matrix $B(*,*,*)$
$b_i^i(*,*)$	$i$ th element of the $i$ th subsystem input matrix $B_i(*,*)$
$\underline{b}_i^i, \bar{b}_i^i$	minimum and maximum bound of $b_i^i(*,*)$ , respectively.
$b_i^i$	$i$ th element of the nominal input matrix $B_i$ for the $i$ th subsystem
$\Delta b_i^i(*,*)$	$i$ th element of the uncertain input matrix $\Delta B_i(*,*)$ for the $i$ th subsystem
$d_i$	distance along $z_{i-1}$ -axis from the origin of the $i$ -1th coordinate frame to the intersection of the $x_i$ and $z_{i-1}$ -axes
$e(*,*)$	'lumped' uncertain elements
$e_{Li}$	'lumped' uncertain elements for the $i$ th subsystem using decentralized local control law
$e_{Gi}$	'lumped' uncertain elements for the $i$ th subsystem using decentralized global control law
$e_i^{L2}$	'lumped' uncertain elements at the lower level for the $i$ th decoupled subsystem using the second hierarchical control law
$e_i^{U2}$	'lumped' uncertain elements at the upper level for the $i$ th subsystem the second hierarchical control law
$f_{Bi}$	element of the load distribution matrix $F_{Bi}$ for the $i$ th actuator
$\bar{g}_{ij}$	upper bound of the norm of the $ij$ inerconnection matrix $A_{ij}(*,*)$



$\bar{g}_{ij}$	gravity row vector
$i_{ai}(*)$	armature current for the $i$ th motor (A)
$i_{fi}(*)$	field current for the $i$ th motor (A)
$k_{ti}$	torque constant for $i$ th motor (Nm/A)
$k_{vi}$	back emf constant for $i$ th motor (V/rad/s)
$l_i$	length of the $i$ th manipulator link (m)
$m_i$	mass of the $i$ th manipulator link (Kg.)
$r_{ij}^i$	bounds on the values of $\Delta a_{ij}^i(*,*)$
$\bar{r}_{ij}$	position vector of the centre of gravity of the $i$ th link with respect to the $i$ th coordinate frame
$s_i^i$	bounds on the values of $\Delta b_i^i(*,*)$
$t$	time (second)
$v_i(t)$	input voltage (armature) for the $i$ th actuator (V)
$v_{fi}(t)$	field voltage of the $i$ th actuator (V)
$w_{Bi}$	element of the rate of the load distribution matrix $W_{Bi}$ for the $i$ th actuator
$x_{pi}$	$i$ th state variable for the manipulator linkage $\equiv i$ th component of the $X_p(t)$ state vector

### 3. Greek Symbols :

$\alpha_i$	twist angle from $z_{i-1}$ -axis to the $z_i$ -axis, measured about the $x_i$ -axis
$\beta_i$	$i$ th positive constant
$\epsilon$	thin boundary layer which decides the size of the switching region for the overall system
$\epsilon_i$	thin boundary layer which decides the size of the switching region for the $i$ th subsystem
$\eta$	radius of the closed ball $\mathfrak{B}$
$\eta_L$	radius of the closed ball $\mathfrak{B}$ for robot manipulator using decentralized local control law
$\eta_G$	radius of the closed ball $\mathfrak{B}$ for robot manipulator using decentralized global control law
$\eta_{H1}$	radius of the closed ball $\mathfrak{B}$ for robot manipulator using the first hierarchical control strategy
$\eta_{H2}$	radius of the closed ball $\mathfrak{B}$ for robot manipulator using the second hierarchical control strategy
$\eta_{Li}$	radius of the closed ball $\mathfrak{B}$ for the $i$ th decoupled subsystem using the second hierarchical control strategy
$\theta(*), \dot{\theta}(*), \ddot{\theta}(*)$	$N \times 1$ vector of joint displacement (rad), joint velocity (rad/s), and joint acceleration (rad/s <sup>2</sup> ), respectively.

$\theta_{di}, \dot{\theta}_{di}, \ddot{\theta}_{di}$	desired joint angle, velocity, and acceleration, respectively for the $i$ th joint.
$\theta_i, \dot{\theta}_i, \ddot{\theta}_i$	$i$ th joint angle, velocity, and acceleration, respectively.
$\theta_{mi}(\cdot)$	angular displacement for the $i$ th motor (rad)
$\underline{\kappa}$	a constant corresponding to the smallest Lyapunov ellipsoid $\mathfrak{X}(\underline{\kappa})$
$\underline{\kappa}_{H2}$	a constant corresponding to the smallest Lyapunov ellipsoid $\mathfrak{X}(\underline{\kappa})$ obtained using the second hierarchical control method
$\lambda(\cdot)$	eigenvalue of $(\cdot)$
$\mu_{Li}(\cdot, \cdot, \cdot)$	'switching surface' for decentralized local controller for the $i$ th subsystem
$\mu_{Gi}(\cdot, \cdot, \cdot)$	'switching surface' for decentralized global controller for the $i$ th subsystem
$\xi$	parameters of the manipulator such as payload which belong to the finite region of allowable parameter values $\Xi$ .
$\rho(\cdot, \cdot)$	norm bound of the 'lumped' uncertainties
$\rho_{Li}(\cdot, \cdot)$	norm bound of the 'lumped' uncertainties for decentralized local control law for the $i$ th subsystem
$\rho_{Gi}(\cdot, \cdot)$	norm bound of the 'lumped' uncertainties for decentralized global control law for the $i$ th subsystem
$\rho_i^{L2}(\cdot, \cdot)$	norm bound of the 'lumped' uncertainties computed at the lower level for the $i$ th decoupled subsystem using the second hierarchical control method

$\rho_i^{U1}(*,*)$	norm bound of the 'lumped' uncertainties computed at the upper level for the $i$ th subsystem using the first hierarchical control method
$\rho_i^{U2}(*,*)$	norm bound of the 'lumped' uncertainties computed at the upper level for the $i$ th subsystem using the second hierarchical control method
$\tau$	time interval for manipulator to move from a given initial position to a final desired position (second)
$\Delta_i$	distance between the final and initial position
$\Xi$	finite region of allowable values for robot parameters, such as payload
$\Phi(*,*)$	nonlinear saturating controller
$\Phi_{Li}(*,*)$	nonlinear control component of the decentralized local controller for the $i$ th subsystem
$\Phi_{Gi}(*,*)$	nonlinear control component of the decentralized global controller for the $i$ th subsystem
$\Phi_i^{L2}(*,*)$	nonlinear control component of the decentralized controller generated at the lower level for the $i$ th subsystem using the second hierarchical control method
$\Phi_i^{U1}(*,*)$	nonlinear control component generated at the upper level for the $i$ th subsystem using the first hierarchical control method
$\Phi_i^{U2}(*,*)$	nonlinear control component generated at the upper level for the $i$ th subsystem using the second hierarchical control method

$\Omega(\ast)$	3x1 vector of open loop control for centralized controller
$\Omega_i(\ast)$	open loop control for the $i$ th decoupled nominal subsystem

# CHAPTER 1

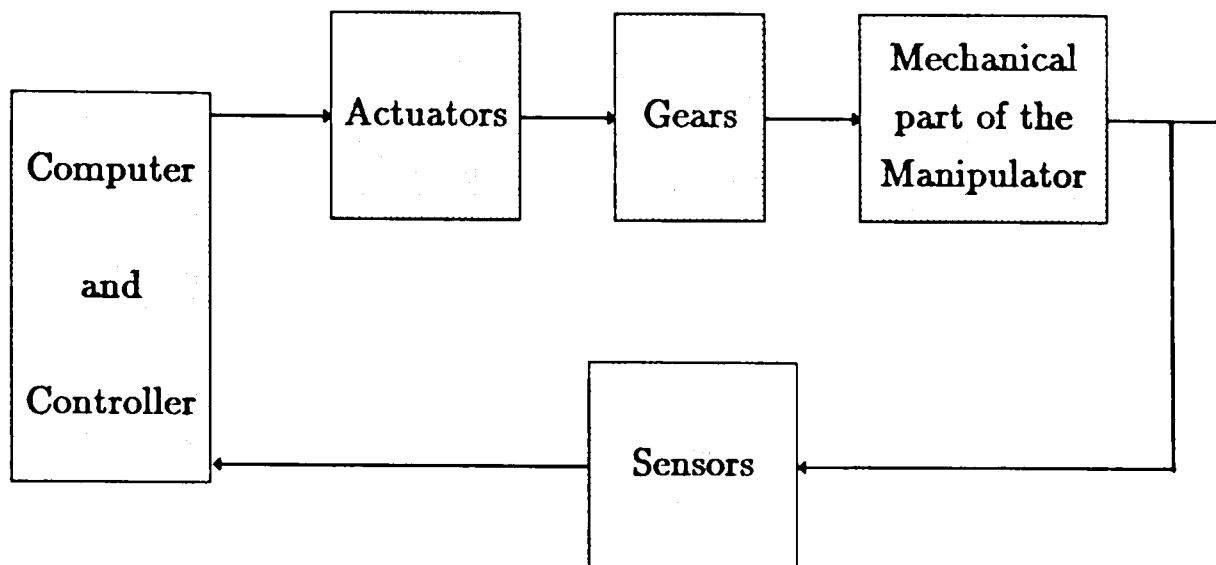
## INTRODUCTION

### 1.1 ROBOTIC SYSTEM

A robot manipulator, which is an outgrowth of the machine tool industry, is a general purpose computer controlled manipulator consisting of several rigid links connected in series by revolute or prismatic joints. One end of the chain is attached to a supporting base while the other end is free and attached with a tool, or end-effector to manipulate objects or perform assembly tasks. A simple end-effector usually has two opposing, moving plates for grasping an object.

Mechanically, a robot manipulator is composed of an arm subassembly, a wrist subassembly, and an end-effector. The arm subassembly and the wrist subassembly, which are the positioning and the orientation mechanisms, respectively, are normally each of three degree of freedom (dof) of movement. Normally, the joints of the manipulator are actuated by either electric or hydraulic actuators through a gearing mechanism. The motion of the joints produced by the actuators determine the position and orientation of the end-effector at any time.

Present day robot manipulators are also equipped with sensors to give accurate measurements of the joint displacements, velocities, and/or accelerations in order to provide information for determining the position and orientation of the end-effector, and to control the motion of the robot manipulator. Furthermore, some manipulators are also equipped with force/torque transducers, tactile sensors or range sensors to provide information about the contact of the end-effector with the environment. Figure 1.1 illustrates the common components of a robotic system.



**FIGURE 1.1 : Robotic System Components**

For many robot manipulator applications, the end-effector of the manipulator is made to move along a desired trajectory, starting from a given initial position and orientation and stop at a specified terminal position and orientation, in the manipulator workspace. A manipulator workspace represents all the possible positions that the end-effector of the robot manipulator can reach. Depending on the particular control technique used, the positions, velocities as well as the accelerations of the generalized coordinates as functions of time are often included in the description of the manipulator trajectory. The trajectory may also include the desired force when the manipulator is in contact with the environment. The desired trajectory for a specific task to be tracked by the robot manipulator may be precomputed off-line in advance and stored in the memory of the controlling computer.

Today, robot manipulators are increasingly being used in a wide variety of manufacturing and technical applications. They are used in hazardous environments such as handling of hazardous (eg. radioactive) materials in nuclear plants, chemical and gas polluted surroundings, and dangerous mining areas. They are also being employed in space expeditions and deep undersea explorations where it is not suitable for human operation. But the main use of robot manipulators is for repetitive tasks which require speed, consistency and accuracy

in a wide variety of automated industrial assembling and fabrications such as material moving and handling, spot/arc welding, metal sheet or fabric pattern cutting, spray painting in the automotive industry, and parts assembly. Such tasks are generally tedious for human operators while they are most suited for mechanical manipulators.

The majority of current robot manipulators used in industrial assembling and manufacturing tasks are equipped with some simple manipulator control systems which prove to be adequate enough for such operations. As the industrial tasks become too complex, involving too many uncertainties or requiring too much flexibility to adapt to a changing environment, these simple manipulator control strategies become ineffective.

As robot manipulators find more and more advanced applications, accurate, fast and versatile manipulation becomes necessary. To achieve higher speed and accuracy for robot manipulators over a wide range of applications, the control technique needs to be improved. In general, the dynamic performance of a robot manipulator is directly dependent on the efficiency of the control algorithm and the dynamic model of the robot manipulator. Thus the robot manipulator control problem consists of determining the mathematical model of the robot manipulator system and then specifying the corresponding control strategies based on these models so that the desired system response and performance is achieved.

## 1.2 MANIPULATOR LINK DYNAMIC MODEL

A number of techniques for developing an efficient analytical model of the mechanical part of a robot manipulator are available. Among these are the Lagrange-Euler method [Lee, 1982; Lee, 1983; Paul, 1981; Vukobratovic and Potkonjak, 1982], the Recursive-Lagrange method [Hollerbach, 1980], the Newton-Euler method [Luh et.al., 1980b; Orin et.al., 1979], and the methods based on the generalized D'Alembert principle [Vukobratovic and Potkonjak, 1982]. All of these methods provide equations which describe the three-dimensional motion of the robot manipulator.



It should be noted that the structure of the equations formulated based on one method may differ from the structure of the equations obtained by another method. However, these equations are 'equivalent' to each other in the sense that they provide the same dynamic response of the robot manipulator.

Of the available methods, the Newton-Euler method and the method based on the Lagrange formulation are the most commonly used approaches for deriving the dynamic equations of the mechanical part of a robot manipulator. The Newton-Euler method yields a set of forward and backward recursive equations, which results in a tremendous reduction of computational time allowing real-time (on-line) control application. However, the Newton-Euler model does not provide sufficient insight for control design due to the recursive computation.

Deriving the dynamic model of a manipulator using the Lagrange-Euler method is simple and systematic. Furthermore, the method gives a set of dynamic equations in a compact matrix form which is appealing from the control viewpoint. However, from the computational point of view, the method is not very convenient and real-time control based on the dynamic model derived is difficult to achieve. To improve the computational efficiency of the Lagrange-Euler method, a Recursive-Lagrange formulation was developed by Hollerbach [1980], but the recursive equations destroy the 'structure' of the dynamic equations which is appropriate for control design purposes.

For an  $N$  dof manipulator as shown in Figure 1.2, using the Denavit-Hartenberg matrix representation (Appendix A) for the manipulator linkage and the Lagrange-Euler formulation as outlined in Appendix B, the dynamic equations describing the motion of the manipulator in the absence of actuator dynamics, friction, and other disturbances can be written in the following matrix form :

$$M(\theta(t), \xi) \ddot{\theta}(t) + D(\theta(t), \dot{\theta}(t), \xi) + G(\theta(t), \xi) = T(t) , \quad (1.1)$$

$$\theta(t) = [ \theta_1(t), \theta_2(t), \dots, \theta_N(t) ]^T$$

$$\theta(t) \in \mathbb{R}^N , \quad \dot{\theta}(t) \in \mathbb{R}^N , \quad \ddot{\theta}(t) \in \mathbb{R}^N$$

where

- $M(\theta(t), \xi)$  :  $N \times N$  inertia matrix  
 $D(\theta(t), \dot{\theta}(t), \xi)$  :  $N \times 1$  vector of Coriolis and centrifugal forces  
 $G(\theta(t), \xi)$  :  $N \times 1$  vector of gravitational forces  
 $T(t)$  :  $N \times 1$  vector of generalized driving forces/torques applied by the actuators at the drive points on each link of the manipulator  
 $\theta(t), \dot{\theta}(t), \ddot{\theta}(t)$  :  $N \times 1$  vectors of generalized joint displacements, velocities, and accelerations respectively  
 $\xi$  : a vector (with appropriate dimension) of parameters of the mechanism such as payload, which belong to the finite region of allowable parameter values  $\Xi$ , that is,  $\xi \in \Xi$ .

In general, the elements of the  $M(\theta(t), \xi)$ ,  $D(\theta(t), \dot{\theta}(t), \xi)$ , and  $G(\theta(t), \xi)$  matrices are functions of the dynamic parameters of the manipulator (lengths, masses, load, etc.) as well as the instantaneous configuration (position) of the links, while the elements of the  $D(\theta(t), \dot{\theta}(t), \xi)$  matrix also depend on the instantaneous velocity of the links. Thus, the elements of these matrices are strongly varying while the manipulator is in motion.

The inertia matrix  $M(\theta(t), \xi)$  is positive definite, bounded, symmetric and always invertible for all  $\theta(t)$ ,  $\xi$ , and  $t$  [Balestrino et.al, 1983; Paul, 1981; Tourasis and Newman, 1985]. The inertia and the gravitational terms are particularly important in manipulator control as they affect the system stability and positional accuracy of the manipulator. It is generally assumed that the Coriolis and centrifugal forces are significant only when the manipulator is moving at high speed. At low speed, their effects are small [Paul, 1981; Tourasis and Newman, 1985].

The Coriolis and centrifugal vector  $D(\theta(t), \dot{\theta}(t), \xi)$  can also be written in the following form [Lim and Eslami, 1985] :

$$D(\theta(t), \dot{\theta}(t), \xi) = \bar{D}(\theta(t), \xi) V(\dot{\theta}(t)) \dot{\theta}(t), \quad (1.2)$$

where

$$V(\dot{\theta}(t)) = \begin{bmatrix} \dot{\theta}_1(t) I_N \\ O_{N-1,1} & \dot{\theta}_2(t) I_{N-1} \\ \vdots & \vdots \\ O_{1,N-1} & \dot{\theta}_N(t) \end{bmatrix}, \quad (1.3)$$

where  $\bar{D}(\theta(t), \xi)$  is a  $N \times \sum_{i=1}^N i$  matrix,  $V(\dot{\theta}(t))$  is a  $\sum_{i=1}^N i \times N$  matrix,  $O_{i,j}$  is an  $i \times j$  null matrix, and  $I_k$  is a  $k \times k$  identity matrix.

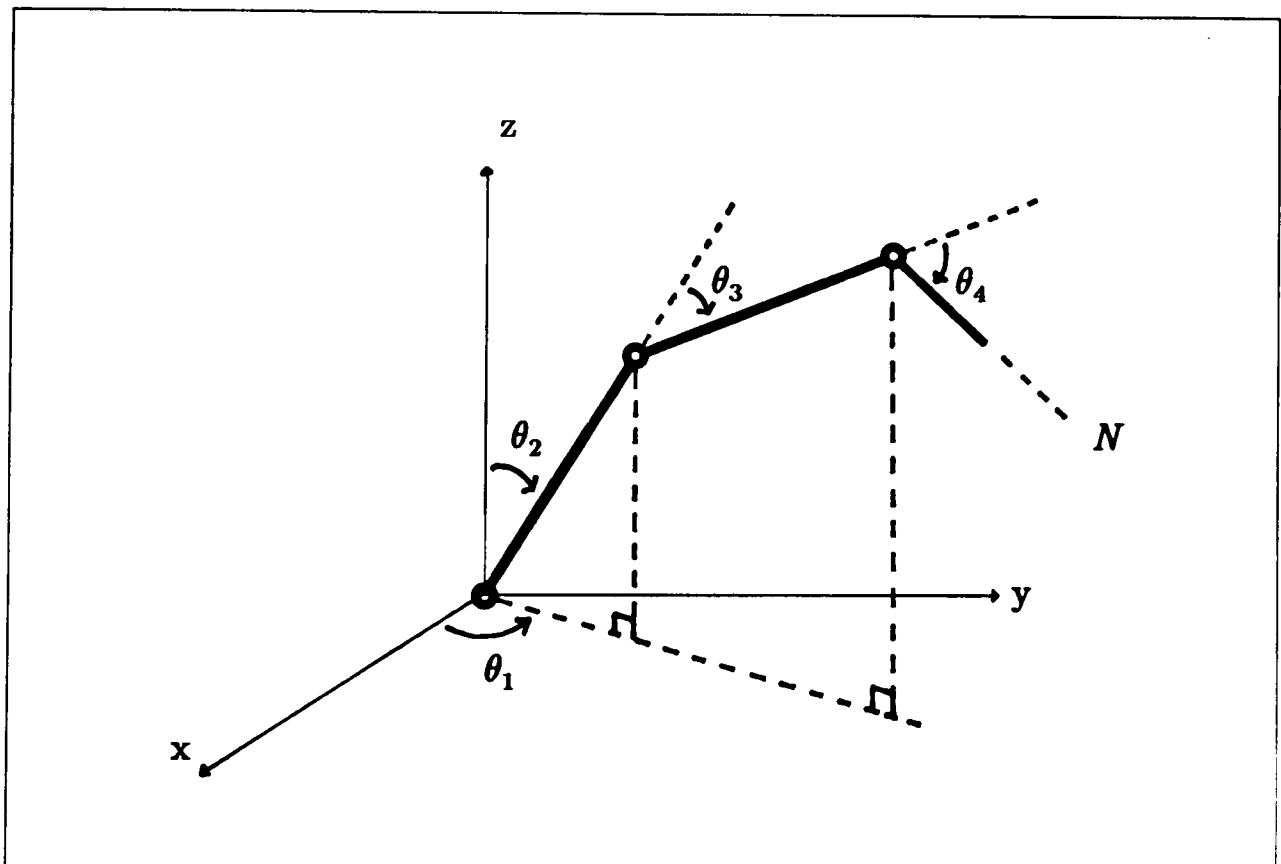


FIGURE 1.2 :  $N$  dof Robot Manipulator

From the above equations, it is clear that the manipulator is a multi-input

multi-output system, described by a set of coupled, nonlinear time varying second-order differential equations, resulting from the dynamic interactions among the links of the manipulator arm. However, in designing the controller for a robot manipulator, the majority of current industrial approaches ignore the coupled, time varying dynamic equations and treat the robot manipulator as a set of simple, decoupled, linear time invariant servomechanisms. However, the changes in the parameters of the controlled system due to the nonlinear time varying dynamics are significant enough to render the linear servomechanism strategy ineffective for fast and high-precision tasks [Fu et.al., 1987; Spong and Vidyasagar, 1989].

In contrast to the current industrial approaches where the controller is normally designed based only on the actuators dynamics, the advanced control strategies proposed in much of the literature on robot manipulator control, were designed based only on the dynamic model of the mechanical part of the robot manipulator (equation 1.1); the dynamics of the actuators which are part of the whole robot manipulator system have generally been ignored, and the drive torques or forces are modeled as pure torque/force sources or as first order lags [Good, 1985]. This, in the majority of cases, is a simplification of a much more realistic model of the system [Ailon, 1988].

### 1.3 MANIPULATOR CONTROL STRATEGIES

The robot manipulator control system is basically organized in a vertical hierarchical fashion which consists of several control levels. The question of how many different levels should there be in the hierarchical structure is dependent on the type of the robot manipulator and the complexity of the tasks for which the robot manipulator is intended. One thing certain is that all robot manipulators have the lowest two control levels, namely, the tactical level which generates the desired trajectories of each dof of the manipulators, and the executive level which executes these trajectories by producing proper drive signals, through a suitable control algorithm, to the appropriate actuators incorporated in each dof of the robot manipulator. It should be noted that each of these levels may be further

subdivided in a number of separate hierarchical levels [Vukobratovic and Stokic, 1982].

Ideally, the aim of robot manipulator controller at the executive level is to maintain a prescribed motion for the manipulator along a desired trajectory generated at the tactical level, and in accordance with some prespecified system performance. The majority of current industrial approaches to the robot arm control design treat each joint of the manipulator as a simple linear servomechanism with, for example, proportional plus derivative (PD), or proportional plus integral plus derivative (PID) controllers. In designing the controllers, the nonlinear, coupled and time-varying dynamics of the mechanical part of the robot manipulator system have usually been completely ignored, or assumed as disturbances. They generally give satisfactory performance when properly tuned and driving only one joint at a time. However, when the links are moving simultaneously and at high speed, the nonlinear coupling effects and the interaction forces between the manipulator links may decrease the performance of the overall system and increase the tracking error. The disturbances and uncertainties such as variable payload in a task cycle may also reduce the tracking quality of the robot manipulator system [Vukobratovic et.al., 1985]. Thus, the method is only suitable for relatively slow manipulator motion and limited-precision tasks [Craig, 1986; Fu et.al. 1987; Spong and Vidyasagar, 1989].

In the past years, various advanced and sophisticated control strategies at the executive level of the control hierarchy have been proposed by numerous researchers for controlling the robot manipulator such that the system is stable as well as the motion of the manipulator arm is maintained along the prescribed path generated by the tactical level. The structures of these controllers can be loosely grouped into three categories, namely, the centralized, decentralized, and multilevel hierarchical structures. In the following, some of these control approaches which have been reported in the literature will be briefly presented.

A number of centralized control schemes have been developed for improving the control of the nonlinear, coupled time varying robot manipulator. These include among others, the Computed Torque techniques [Markiewicz, 1973; Bejczy, 1974; Craig, 1986] or the Inverse Problem methods [Raibert and Horn, 1978; Paul, 1981], the Resolved Motion Control strategies [Whitney, 1972; Luh

et.al., 1980a; Wu and Paul, 1982], the Nonlinear Decoupled Control methods [Freund, 1982; Kuo and Wang, 1989], the Variable Structure Control approaches [Young, 1978; Slotine and Sastry, 1983; Chen et.al., 1990; Fu and Liao, 1990; Wijesoma and Richards, 1990; Song and Gao, 1991], the Self-Tuning Control methods [Koivo and Guo, 1983; Leininger, 1984], and the Model Reference Adaptive Control strategies [Dubowsky and DesForges, 1979; Balestrino et.al., 1983; Osman, 1985; Tzafestas and Stavrakakis, 1986; Chen, 1987b; Ambrosino et.al., 1988].

The Computed Torque Technique [Markiewicz, 1973; Bejczy, 1974; Craig, 1986], also called the Inverse Problem method [Raibert and Horn, 1978; Paul, 1981], is basically composed of a feedforward and a feedback component. The methods require exact modeling of the manipulator arm dynamics. Based on the manipulator dynamic model and the desired joint trajectory, the feedforward component computes the required input torques to compensate the actual manipulator arm dynamics such that the nonlinear coupled manipulator system is reduced into a linear decoupled time invariant system. Then, based on a standard linear control technique, the feedback component is designed to control the linear decoupled system. One of the main drawbacks of these control methods [Markiewicz, 1973; Bejczy, 1974; Paul, 1981; Craig, 1986] is that they require on-line computation of the joint torques based on the complete Lagrange-Euler dynamic equation of the robot manipulator, which is very inefficient. Raibert and Horn [1978] used a partial table look-up approach to simplify the computation automatically in the control computer rather than to compute the nonlinear dynamic equations. The disadvantage of this scheme is that it requires a large computer memory. Another problem with these methods is that their validity is questionable when there is a significant difference between the computed torque model parameters and the actual robot manipulator parameters.

Resolved motion means that the motion of the various joints are combined and resolved into separately controllable end-effector motion along the task coordinates axes (Cartesian coordinate system) [Fu et.al, 1987]. The Resolved Motion Control strategies include the Resolved Motion Rate Control [Whitney, 1972], Resolved Motion Acceleration Control [Luh et.al., 1980a], and Resolved Motion Force Control [Wu and Paul, 1982]. In Resolved Motion Rate Control [Whitney, 1972], the joint motors are required to move simultaneously at different

time varying speeds along the axes relevant to the task coordinate in order to achieve desired coordinated end-effector motion. The method requires the computation of the inverse of the Jacobian matrix at each sampling time for the calculation of the joint rate (velocity). The disadvantages of this method are that the inverse Jacobian matrix requires intensive computations, and matrix inversion may cause singularity problems. The Resolved Motion Acceleration Control method [Luh et.al., 1980a] is simply an extension of the Resolved Motion Rate Control technique, which includes acceleration control. It assumes that the desired positions, velocities and accelerations are available. The method is quite similar to the Computed Torque Technique except that the desired trajectory and the feedback law are expressed in terms of the task coordinates. This control technique has the same disadvantages as the Revolved Motion Rate Control technique and the Computed Torque technique. Resolved Motion Force Control [Wu and Paul, 1982] is based on the relationship between the forces obtained from wrist force sensors and the joint torques at the joint actuators. The controller consists of the Cartesian position control which calculates the desired forces to be applied to the end-effector in order to track the desired task trajectory, and a force convergent control which determines the required joint torques to be applied to each joint actuator such that the end-effector has the desired forces applied as determined by the Cartesian position control. The method avoids the computation of the inverse Jacobian matrix and complicated dynamics of the robot manipulator. However, the convergent rate of the force convergent control depends on the structure of the manipulator, and the calculation of the dynamic model is replaced by an iterative stochastic method which also demands extensive computation [Vukobratovic et.al., 1985].

A nonlinear feedback pole placement control method for controlling a robot manipulator was proposed by Freund [1982]. The control algorithm consists of two parts. The first part is the nonlinear decoupling control which completely decouples the nonlinear robot manipulator dynamics into a set of decoupled linear input-output second order differential equations whose characteristic coefficients can be chosen arbitrarily. The second component of the Nonlinear Control Algorithm is an arbitrary pole placement technique to design the dynamics of the linear input-output second order equations, obtained from the first part, as desired. It was acknowledged that [Freund, 1982] the method uses a relatively

complicated nonlinear operator and the nonlinear decoupling theory is extremely difficult, especially for robot manipulators with three dof or more, and with complicated dynamic equations. Kuo and Wang [1989] combined the nonlinear state feedback control law of Freund [1982] to obtain a set of decoupled equations in joint coordinates, with a robust servomechanism theory to calculate the linear control signal for each joint of the manipulator to suppress the effects of modeling errors, disturbances and other uncertainties. The nonlinear control part of the control algorithm uses the Newton-Euler recursive formulation to calculate the complete manipulator dynamics on-line.

Several control algorithms for robot manipulators based on the theory of Variable Structure Systems have been reported in the literature [Young, 1978; Slotine and Sastry, 1983]. Variable Structure Systems are characterized by a discontinuous feedback control law on an appropriate switching surface in the state space. The control law induces the sliding mode in which the system trajectories lie on the switching surface, which results in insensitivity to parameter variations and disturbances. It is this insensitivity property that enables the elimination of the interactions among the joints of the robot manipulator. The control algorithm does not require an accurate knowledge of the physical parameters of the manipulator; the bounds of the parameters are sufficient to construct the controller [Young, 1978]. However, the discontinuous control law is very difficult for realization in practice and the discontinuous controller results in chattering effects of the control signal. To reduce the chattering effects due to the discontinuous control inputs, Slotine and Sastry [1983] replaced the discontinuous feedback control law with an approximated continuous feedback control law. However, there is a trade-off between robustness to high frequency dynamics (chattering) and the tracking precision of the robot manipulator. Fu and Liao [1990] constructed a robust tracking controller for robot manipulators based on nonlinear state feedback control theory and Variable Structure control theory. The purpose of the nonlinear control law is to linearize and decouple the system in order to obtain a set of linear output equations, whereas the aim of the Variable Structure Control law is to compensate for the uncertainties and at the same time to provide an asymptotic tracking force. It was shown through simulations that the output of the closed loop robot manipulator system asymptotically tracks the desired trajectories despite the presence of the uncertainties. A robust trajectory



tracking control for a robot manipulator based on the combination of the Computed Torque technique and the Variable Structure theory was proposed by Wijesoma and Richards [1990]. The computed torque component which is based on the nominal dynamic model of the manipulator is used to linearize and decouple the system. Then the Variable Structure theory is used to synthesize a discontinuous control component to eliminate any residual coupling and nonlinear effects due to variations of the system from its nominal dynamics. An almost similar approach was used by Chen et.al. [1990] and Song and Gao [1991] in their design. However, a great amount of on-line computation is needed during the manipulator motion due to the computation of the nonlinear state feedback control law or the nominal torque required for each manipulator joint.

A great deal of research has been conducted towards developing an adaptive robot controller which leads to high-level performance of the robot manipulator in the presence of large variations in the dynamic characteristics of the manipulator, such as load variations and parameter uncertainties. These include the Self-Tuning Control methods [Koivo and Guo, 1983; Leininger, 1984,], and The Model Reference Adaptive Control (MRAC) techniques [Dubowsky and DesForges, 1979; Balestrino et.al., 1983; Nicosia and Tomei, 1984; Osman, 1985; Chen, 1987b; Ambrosino et.al., 1988].

The adaptive Self-Tuning Control proposed by Koivo and Guo [1983] used an autoregressive model to fit the input-output data from the robot manipulator. The method does not require a detailed mathematical model of the robot manipulator, and the resulting system is insensitive to the changing configurations of the manipulator and variations in the load. However, the dynamic model of the manipulator is required to be linearized. Thus, the technique results in a poor system performance over a wide range of tasks. Leininger [1984] proposed a self-tuning pole placement method which provides an adaptive feedback design approach which does not requires an a priori mathematical description of the robot manipulator dynamics. The method automatically compensates for the manipulator compliance, friction, and link flexibility through the on-line learning mechanism. However, the behaviour of the manipulator when starting the learning process depends mostly on the accuracy in initialization of the prediction model parameters. This accuracy determines the degree of 'erratic' motion at task start-up. Another problem associated with the method is that the

convergence of the parameter estimation and controller gains may not be achieved during the finite time over which the motion takes place [Vukobratovic et. al., 1985].

The MRAC technique is based on the selection of an appropriate reference model which specifies the design objective (rise time, delay time, etc.), and an adaptation mechanism which modifies the feedback gains to the system. The objective of the control system is to minimize the error between the states or outputs of the reference model and those of the actual robot manipulator via the chosen adaptation mechanism. Normally, linear second-order time invariant reference models were used for each dof of the robot manipulator. Dubowsky and DesForges [1979] used the steepest descent method to adjust the gains in the position and velocity feedback loops. The method is computationally less burdensome than the methods which evaluate a complete nonlinear robot manipulator model, and it has good noise rejection properties [Vukobratovic et.al., 1985]. However, the interaction forces among the joints of the manipulator are assumed to be negligible and consequently ignored. Hence, the method does not result in satisfactory control particularly for the faster and more sophisticated tasks. Several Adaptive Model Following Control (AMFC) techniques that nullify the difference between the behaviour of the robot manipulator and the reference model via Popov's hyperstability theory [Balestrino et.al., 1983; Nicosia and Tomei, 1984], and the Variable Structure theory [Osman, 1985] have been proposed. The methods require perfect model matching conditions to be satisfied. The control algorithms are insensitive to parameter variations and disturbances, and force the nonlinear, coupled time varying robot manipulator system to have well-behaved, linear, uncoupled characteristics. It is, however, difficult to establish any stability analysis of the controlled system [Fu et. al., 1987], and the techniques are complicated mathematically due to the centralized nature of the control approaches [Vukobratovic et.al., 1985].

Eventhough some of the aforementioned centralized methods, notably the MRAC and the Variable Structure Control techniques, exhibit excellent and promising results in simulation or in a laboratory environment, very few results could be transferred into practice. This is due to the fact that the centralized control schemes, in general, require excessive computation time, are complex and are costly to implement. The centralized approach treats the robot manipulator

as a single plant which is unfavourable and impractical from the view point of reliability, implementation and maintenance of the controllers.

Furthermore, most of the above control strategies were developed based on a model of the robot manipulator (i.e. equation 1.1) that is inadequate because the actuators dynamics have not been taken into account. Normally, the drive torques/forces are modelled as pure torque/force sources or as first order lags [Good, 1985]. Since the actuators are part of the robot manipulator system, the introduction of their dynamic behaviour is essential for a more realistic presentation of the complete dynamics of the robot manipulator, and for the synthesis of high performance control algorithms.

Eventhough the actuator dynamics constitute an important part of the complete robot manipulator system, only a limited amount of research on robot manipulator controller synthesis based on the complete dynamic model of the system can be found in the literature. A MRAC technique and hyperstability approach similar to that of Nicosia and Tomei [1984] was used by Tzafestas and Stavrakakis [1986] to synthesize a robot manipulator controller based on the complete dynamic model of the manipulator and its actuators (DC motors). The method is unnecessarily complicated due to the need to find the reflected electromagnetic torque which guarantees the desired trajectory, velocity, and acceleration, and for inclusion in the reference model. Beekmann and Lee [1988] applied Freund's nonlinear control theory [1982] to decouple the robot manipulator model that contains the link dynamics, motor dynamics, and the interaction dynamics. A pole placement method was then applied to control the decoupled system. The derivation of the control law is complicated and requires extensive on-line calculation of the nonlinear controller. A similar problem arises with the control method proposed by Tarn et.al. [1991]. A nonlinear decoupled control law was designed to linearize and decouple the complete model of the robot manipulator and its actuator into a number of decoupled linear subsystems in the task space. An optimized linear controller was then designed to render the system robust against system parameter uncertainties. The controller requires the computation of the complete dynamic model of the robot manipulator system on-line, which is a drawback. While the above studies were based on a robot manipulator actuated with DC motors, the study of the influence of electrohydraulic actuators with various degree of complexities, on the synthesis of

a robot manipulator controller can also be found in the literature [Katic and Vukobratovic, 1986].

While it is necessary to consider the effect of the actuator dynamics in synthesizing the robot manipulator controller, the inclusion of the servomechanism dynamics into the robot manipulator dynamic equations will inevitably increase the order and complexity of the overall dynamic model of the system. Accordingly, the control law required to control the system will become more complex, particularly if the controller structure is in the centralized form.

In order to circumvent the problems due to the centralized structure of the control system at the executive level, it is necessary to decentralize the control effort or to decompose the executive control level further into a two-level hierarchical control structure.

#### 1.4 RESEARCH OBJECTIVES

The objectives of this research are as follow :

- A. To give a unifying framework for the formulation of the complete mathematical dynamic model of a DC motor actuated revolute robot manipulator in state variable form. The formulations result in nonlinear time varying state equations which are believed to represent a more realistic model of the robot manipulator than the model with the drive torques/forces modelled as ideal pure torque/force sources or as first order lags, and to provide a better model for advanced controller design purposes. Two different structures of the model are outlined where each model is based on a different set of state variables. The first set of the state variables consists of the manipulator joint angle, velocity and the motor armature current, while the manipulator joint angle, velocity, and acceleration forms the second set of the state vector. It will be shown that the integrated robot manipulator model with the joint position, velocity and acceleration as the state variables is more suitable for the synthesis of high performance control algorithms.

- B. To decompose and transform the integrated nonlinear dynamic model of the robot manipulator into a set of interconnected subsystems with bounded uncertainties.
- C. To synthesize decentralized tracking controllers for a robot manipulator based on a deterministic approach. Two tracking control algorithms are proposed : local decentralized control and global decentralized control. The local decentralized controller utilizes only the local states of the subsystem, while the global decentralized control law uses the local states as well as the states of the other subsystems as feedback information. Both methods assume that the bounds on the nonlinearities and uncertainties present in the system are known. It will be shown that in both methods, the resulting errors between the responses of the actual robot manipulator system and that of the reference trajectories are uniformly ultimately bounded with respect to any arbitrarily small set of ultimate boundedness; in spite of the highly nonlinear and coupled robot manipulator dynamics, and the uncertainties present in the system.
- D. To formulate two-level hierarchical controllers for tracking control of a robot manipulator based on a deterministic approach. Two methods are proposed. In both methods, the control laws are decoupled at the lower level of the hierarchy, and utilize only the local states as the feedback information. It will be shown that the hierarchical controllers will render the nonlinear robot manipulator practically stable and track the desired trajectory within a particular bounded neighbourhood of the trajectory after a finite time.

Verification of the proposed control algorithms are performed through stability analysis using Lyapunov's second method and computer simulation studies using a three dof revolute robot manipulator actuated with DC motors.

## 1.5 STRUCTURE AND LAYOUT OF THESIS

Chapter 2 deals with the formulation of the integrated dynamic models of revolute robot manipulators. First, the state space representation of the dynamic model of the mechanical part of the robot manipulator are outlined. Then, the state space descriptions of the actuator dynamics (DC motor) with two different sets of state variables are presented. The first set of the state variables consists of the joint angle  $\theta(t)$ , the joint velocity  $\dot{\theta}(t)$ , and the armature current  $i_a(t)$ , while  $\theta(t)$ ,  $\dot{\theta}(t)$ , and  $\ddot{\theta}(t)$ , the joint angle, velocity, and acceleration respectively, form the second set of the state variables. Based on the actuator dynamic models, two integrated dynamic models of the robot manipulator in state space description are presented. The advantages and disadvantages of each method is also discussed. Finally, a detailed derivation of the integrated dynamic model for a three dof revolute robot manipulator with  $\theta(t)$ ,  $\dot{\theta}(t)$ , and  $\ddot{\theta}(t)$  – the joint angle, joint velocity, and joint acceleration respectively – as the state variables is presented.

Chapter 3 establishes the basis for the synthesis of the controller based on the deterministic approach presented in the next few chapters. The integrated nonlinear dynamic model of the robot manipulator is decomposed into an input decentralized form. Based on the known allowable range of operation of the robot manipulator and the maximum allowable load, the decomposed model is then transformed into a set of interconnected linear subsystems with bounded uncertainties description.

Chapter 4 outlines a decentralized control strategy for robot manipulators based on a deterministic approach. A brief review of the existing decentralized control techniques and deterministic control approaches for controlling a robot manipulator are given in the earlier part of the chapter. The formulation of the decentralized tracking control problem and some standard assumptions are given next. The method is designed based only on the local states and the bound on the uncertainties as the feedback information. Finally, the performance of the proposed decentralized tracking control approaches is evaluated by means of computer simulation study.

In Chapter 5, a decentralized global tracking control law is proposed. The method uses the local states, the bound on the uncertainties, as well as the states

of the neighbouring subsystems as the feedback information. To evaluate the performance of the controller, the decentralized global controller is then applied to the three dof revolute robot manipulator model derived previously and simulated on a digital computer. Various simulation results are presented and discussed to study and verify the performance of the proposed controllers.

Chapter 6 presents the hierarchical control concepts for robot manipulator control. General multilevel and hierarchical control concepts, and the existing hierarchical control techniques for robot manipulators, are briefly reviewed. Two hierarchical control concepts for robot manipulators based on a deterministic approach are proposed. The simulation results for various case studies are presented and discussed.

Chapter 7 summarizes the results of the studies. Suggestions for future work are also presented at the end of the chapter.

Four Appendices are included. Appendix A presents an algorithm for establishing the coordinate frames and Denavit-Hartenberg (D-H) transformation matrices for robot manipulators. The second appendix outlines the Euler-Lagrange formulation to derive the dynamic model of the mechanical linkage of a robot manipulator. Appendix C is an overview of the existing methods of deriving the complete robot manipulator mathematical model. The last appendix presents the detail components of a three dof revolute robot manipulator dynamic equation.

## CHAPTER 2

### MODELING OF ROBOT MANIPULATOR

#### 2.1 INTRODUCTION

An important initial step in the design of controllers for an industrial robot is to obtain a complete, and as accurately as possible, the mathematical model of the robot manipulator. Such a model is useful for computer simulation of the robot manipulator motion, for the synthesis of a control algorithm for controlling the robot manipulator, etc. As mentioned in the previous chapter, typical industrial robots can be modelled as an open kinematic chain of  $N$ -rigid bodies or links, connected in series by  $N$  joints. Normally, the joints are actuated by either electric or hydraulic actuators.

In much of the literature on advanced control strategies proposed to improve the performance of robot manipulators control algorithms were designed based only on the dynamics of the mechanical links (equation 1.1), where the joint torques/forces are considered as the inputs or the control variables to the system. However, the dynamics of the mechanical part of the robot manipulator alone is not sufficient to represent the dynamics of the robot manipulator, since it does not include the dynamics of the actuating mechanism which generates the joint torques/forces. In fact, it is the input to the actuators, but not the output (joint torques/forces) that is directly controlled [Tarn et.al., 1988].

Since the actuators are part of the robot manipulator system, it is necessary to consider the effects of the actuator dynamics especially in cases where higher speed and better system performance are required [Ailon, 1988; Tarn et.al., 1988]. Hence, it is necessary to include the actuators dynamics into the robot manipulator dynamic equations.



The aim of this chapter is to present the formulation of the integrated mathematical dynamic model of an electrically driven revolute robot manipulator in state variable form. The integrated model comprises the dynamic model of the mechanical links of the robot manipulator as well as the actuators dynamics (permanent magnet DC motors). Two different approaches are presented. In the first approach, the joint angles, velocities and the armature currents of the actuating mechanisms are chosen as the state variables. In the second approach, the armature current is replaced by the joint acceleration as the state variables. The formulations result in third-order nonlinear time varying state equations, which represent a more realistic model of the robot manipulator than the model with the joint torques/forces modeled as pure torque/force sources or as first-order lags. The advantages and disadvantages of the methods will also be discussed. Finally, an integrated model of a three dof revolute robot manipulator driven by DC motors is derived based on the second approach.

It should be noted that, the integrated dynamic models of the robot manipulator derived by no means represent a complete model of the robotic system since the drive system nonlinearities such as Coulomb friction, backlash, stiffening spring characteristic of the actuators, and various sources of flexibility (such as deflection of the links under load and vibrations, elastic deformation of bearings and gears) are not included in the formulation of the integrated dynamic model. However, it is believed that the integrated model derived represents more closely the dynamic behaviour of the robot manipulator, and provides a better and much more suitable model for the purpose of dynamic analysis and advanced controller synthesis for the robot manipulator.

## 2.2 STATE SPACE REPRESENTATION OF MANIPULATOR LINK DYNAMICS

In this section, the dynamic equations of the mechanical part of the robot manipulator are rewritten in the state variable form, as outlined by Balestrino et.al. [1983], and Lim and Eslami [1985; 1986]. This formulation is required in deriving the integrated model of the robot manipulator based on the first

approach.

From equations (1.1) and (1.2), the dynamic equation of the mechanical links of the robot manipulator can be rewritten as follows :

$$M(\theta(t), \xi) \ddot{\theta}(t) + \bar{D}(\theta(t), \xi) V(\dot{\theta}(t)) \dot{\theta}(t) + G(\theta(t), \xi) = T(t). \quad (2.1)$$

In the following, some of the arguments are sometimes suppressed for convenience. Let the state variables for the  $N$  dof manipulator be :

$$\begin{aligned} x_{pi} &= \theta_i \\ x_{pi+N} &= \dot{x}_{pi} = \dot{\theta}_i, \\ i \in \mathcal{I} &\triangleq \{ i : i = 1, 2, \dots, N \}, \end{aligned} \quad (2.2)$$

and, hence, the  $2N$ -dimensional state vector is given as :

$$\begin{aligned} X_p(t) &= \left[ \theta^T(t), \dot{\theta}^T(t) \right]^T, \\ X_p(t) &\in \mathbb{R}^{2N \times 1}. \end{aligned} \quad (2.3)$$

Let the gravitational force vector be represented by

$$G(x_{pi}, \xi) = \hat{G}(x_{pi}, \xi) \begin{bmatrix} x_{p1} \\ \vdots \\ x_{pN} \end{bmatrix}, \quad (2.4)$$

where the matrix  $\hat{G}(x_{pi}, \xi) \in \mathbb{R}^{N \times N}$  is not unique [ Lim and Eslami, 1985; 1986].

Then, in terms of the state variables, equation (2.1) becomes :

$$\dot{X}_p(t) = A_p(X_p, \xi, t) X_p(t) + B_p(X_p, \xi, t) T(t) \quad (2.5)$$

where

$$A_p(X_p, \xi, t) = \begin{bmatrix} O_{NN} & : & I_N \\ \dots\dots\dots & : & \dots\dots\dots \\ A_{p1}(X_p, \xi, t) & : & A_{p2}(X_p, \xi, t) \end{bmatrix} \quad (2.6)$$

$$A_{p1}(X_p, \xi, t) = - M^{-1}(x_{pi}, \xi, t) \hat{G}(x_{pi}, \xi, t) \quad (2.7)$$

$$A_{p2}(X_p, \xi, t) = - M^{-1}(x_{pi}, \xi, t) \bar{D}(x_{pi}, \xi, t) V(\dot{x}_{pi}), \quad (2.8)$$

$$B_p(X_p, \xi, t) = \begin{bmatrix} O_{NN} \\ \dots\dots\dots \\ M^{-1}(x_{pi}, \xi, t) \end{bmatrix}, \quad i \in \mathcal{J}, \quad (2.9)$$

and  $I_N$  is an  $N \times N$  identity matrix, while  $O_{NN}$  is an  $N \times N$  null matrix.

Each element of the matrices (2.6) and (2.9) is a nonlinear function of the state variables, taking into account the contribution of the inertia matrix, Coriolis, centrifugal, and gravitational forces.

## 2.3 ACTUATOR DYNAMICS

For robot manipulators, permanent magnet D.C. motors and electrohydraulic actuators are widely used as the actuating mechanism. Here, an electrically driven manipulator is considered, where each joint of the robot manipulator is driven by an armature controlled permanent magnet DC motor.

For  $i$ th joint, the schematic diagram of a permanent magnet armature controlled DC motor can be illustrated as in Figure 2.1. By the application of Kirchhoff's voltage law to the armature circuit of the motor and Newton's law of

motion to the rotating system, the dynamics of the actuator can be obtained as follows :

$$J_{mi} \ddot{\theta}_{mi}(t) = - B_{vi} \dot{\theta}_{mi}(t) + k_{ti} i_{ai}(t) - T_{Li}(t) \quad (2.10)$$

$$L_i \dot{i}_{ai}(t) = - k_{vi} \dot{\theta}_{mi}(t) - R_i i_{ai}(t) + v_i(t) \quad (2.11)$$

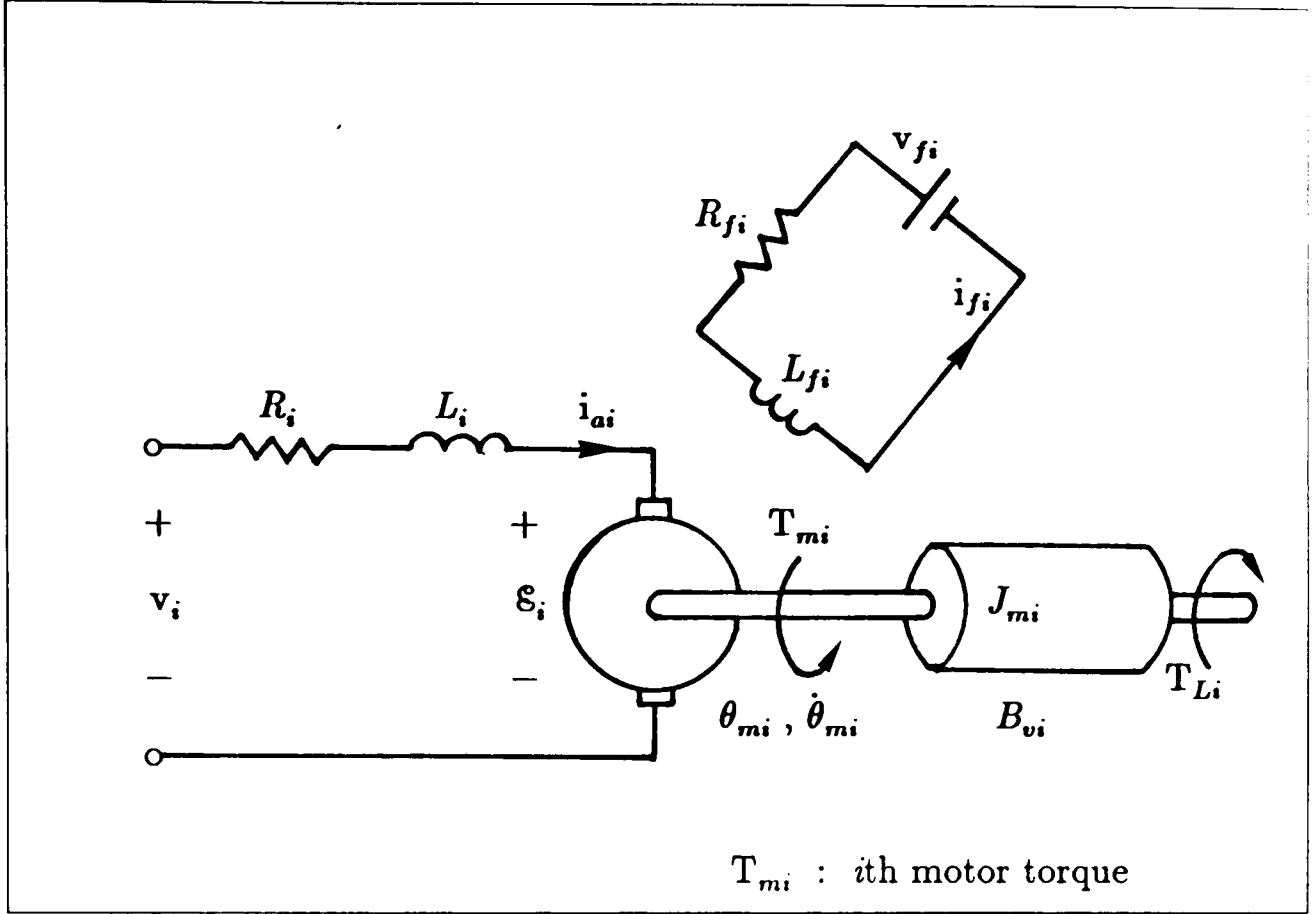
where

- $J_{mi}$  : moment of inertia for  $i$ th motor ( Kgm<sup>2</sup>)
- $\theta_{mi}(t)$  : angular displacement for  $i$ th motor (rad)
- $B_{vi}$  : viscous friction coefficient for  $i$ th motor (Nm/rad/s)
- $k_{ti}$  : torque constant for  $i$ th motor (Nm/A)
- $k_{vi}$  : back emf constant for  $i$ th motor (V/rad/s)
- $L_i$  : armature inductance for  $i$ th motor (H)
- $R_i$  : armature resistance for  $i$ th motor ( $\Omega$ )
- $i_{ai}(t)$  : armature current for  $i$ th motor (A)
- $T_{Li}(t)$  : load torque for  $i$ th motor (Nm)
- $v_i(t)$  : voltage input to the  $i$ th actuator (V)

Notice that the motor dynamics (equations 2.10 and 2.11) are linear and time invariant.

In most of the present robot manipulators, the motor shaft is mechanically coupled to the manipulator link (load) through gears as shown in Figure 2.2. The shaft of the  $i$ th gear train is directly connected to the axis of the  $i$ th motor, and is referred to as the primary side. The output shaft of the gear is on the secondary side, and is coupled to the  $i$ th link of the robot manipulator. Depending on the type or structure of the gear train, the motion of the output shaft may rotate in the same direction as the input shaft or in the opposite direction. Assuming that the gear train is ideal, that is, assuming the gear train is frictionless, there by creating no power losses, and inertialess; the relationship between the loading torque  $T_i(t)$  on the secondary side and the load torque  $T_{Li}(t)$  on the motor shaft at the primary side can be obtained as

$$T_{Li}(t) = \frac{T_i(t)}{N_i} , \quad N_i \geq 1 \quad (2.12)$$



**FIGURE 2.1 : Schematic Diagram Of Armature Controlled Permanent Magnet DC Motor.**

where  $N_i$  is the inverse of the gear ratio. Thus the loading torque  $T_i(t)$  is reduced when it is reflected to the shaft of the motor when  $N_i$  is large.

From equation (2.12), the actuator dynamics then become as follows :

$$J_{mi} \ddot{\theta}_{mi}(t) = -B_{vi} \dot{\theta}_{mi}(t) + k_{ti} i_{ai}(t) - \frac{T_i(t)}{N_i} \quad (2.13)$$

$$L_i \dot{i}_{ai}(t) = -k_{vi} \dot{\theta}_{mi}(t) - R_i i_{ai}(t) + v_i(t) \quad (2.14)$$

The loading torque  $T_i(t)$  acting on the  $i$ th actuator is given by the  $i$ th element of the vector  $T(t)$  of equation (2.1), that is, the dynamics of the mechanical links of the robot manipulator. The inclusion of the load torque  $T_i(t)$

in equation (2.13) will turn the otherwise linear, decoupled, and time invariant actuator dynamics into a nonlinear, coupled, and time varying one. The effect of the nonlinear dynamics of the mechanical links on the actuator dynamics depends very much on the speed of the robot manipulator and the size of the gear ratio. For a slow robot manipulator (and normally with large value of  $N_i$ ), the influence of the nonlinear, coupled, and time varying dynamics of the mechanical links are small and thus, can be ignored. For a fast manipulator and direct drive robot manipulator ( $N_i=1$ ), the influence of the nonlinear dynamics are very significant and must be taken into consideration.

In-order to reflect the actuator dynamics to the manipulator side of the gearing mechanism, the following identity is used :

$$\theta_{mi}(t) = N_i \theta_i(t) \tag{2.15}$$

Then, after rearranging, the equations (2.13) and (2.14) become

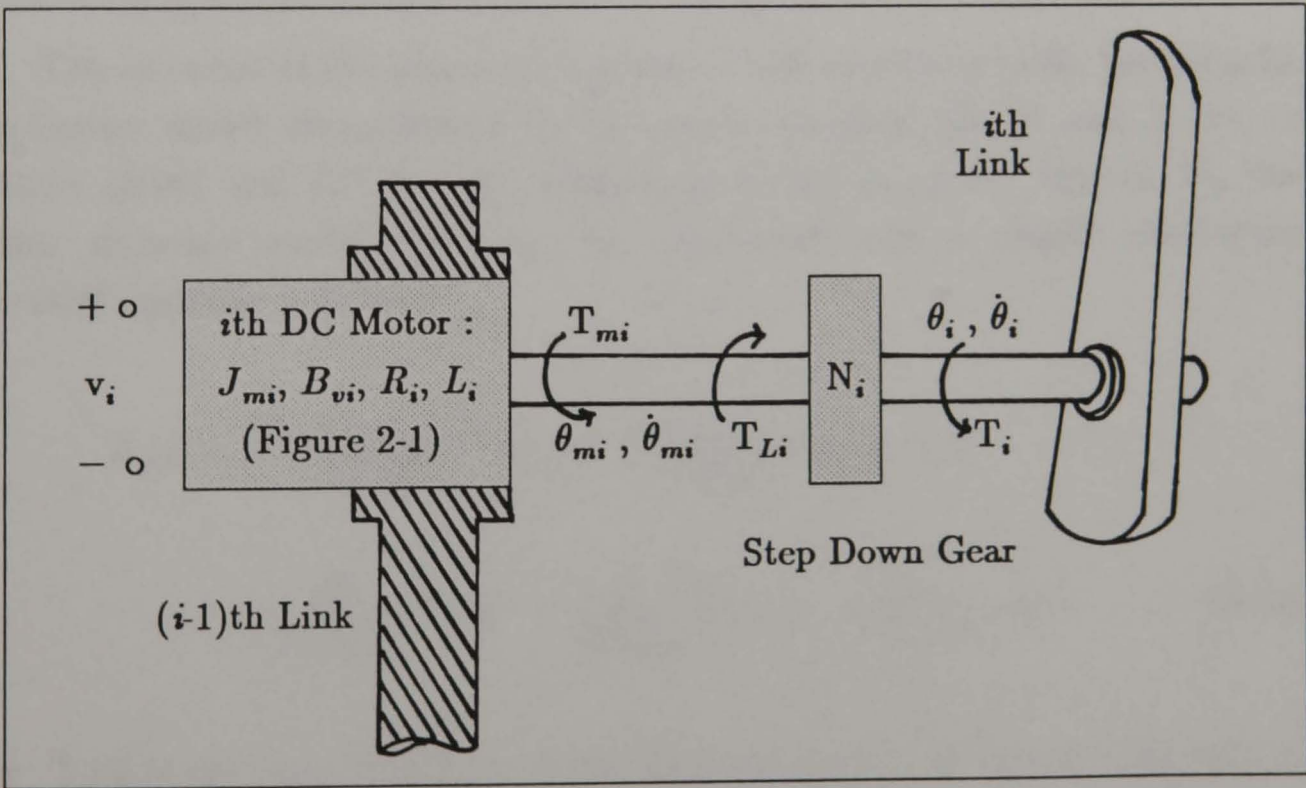


FIGURE 2.2 : Diagram Of The  $i$ th Manipulator Link Actuated by DC Motor With Gear Train.

$$\ddot{\theta}_i(t) = -\frac{B_{vi}}{J_{mi}} \dot{\theta}_i(t) + \frac{k_{ti}}{N_i J_{mi}} i_{ai}(t) - \frac{1}{N_i^2 J_{mi}} T_i(t) \quad (2.16)$$

$$\dot{i}_{mi}(t) = -\frac{N_i k_{vi}}{L_i} \dot{\theta}_i(t) - \frac{R_i}{L_i} i_{ai}(t) + \frac{1}{L_i} v_i(t) . \quad (2.17)$$

If the armature inductance is small (i.e. if the electrical time constant of the motor  $L_i/R_i$ , is negligible), the inductance  $L_i$  can be ignored. This will reduce the actuator to a second-order model described by the following second-order differential equation :

$$\ddot{\theta}_i(t) = -\left[ \frac{B_{vi}R_i + k_{vi}k_{ti}}{J_{mi}R_i} \right] \dot{\theta}_i(t) + \frac{k_{ti}}{N_i J_{mi}} v_i(t) - \frac{1}{N_i^2 J_{mi}} T_i(t) \quad (2.18)$$

In the cases where the viscous friction constant of the motor,  $B_{vi}$ , can be assumed negligible, the actuator viscous friction term  $B_{vi}\dot{\theta}_i(t)$  can also be ignored.

The inclusion of the armature inductance will inevitably make the actuator a third-order model characterized by the above equations (2.10) and (2.11), or equations (2.16) and (2.17). By combining equations (2.16) and (2.17), the actuator dynamic model can also be represented by a single third-order differential equation as follows :

$$\ddot{\theta}_i(t) + \frac{(B_{vi}L_i + J_{mi}R_i)}{J_{mi}L_i} \dot{\theta}_i(t) + \frac{(k_{vi}k_{ti} + B_{vi}R_i)}{J_{mi}L_i} \theta_i(t) + \frac{R_i}{N_i^2 J_{mi}L_i} T_i(t) + \frac{1}{N_i^2 J_{mi}} \dot{T}_i(t) = \frac{k_{ti}}{N_i J_{mi}L_i} v_i(t) . \quad (2.19)$$

where  $\dot{T}_i(t)$  is the time derivative of the load torque  $T_i(t)$  due to the  $i$ th joint of the manipulator on the  $i$ th motor.

In this research, a third-order actuator model is considered. In the following, two different forms of actuator dynamics in state space description are outlined. The first form is based on equations (2.16) and (2.17) where the joint

position, joint velocity, and armature current are chosen as the state variables, while equation (2.19) form the basis of the second approach where the joint position, velocity, and acceleration are chosen as the state variables.

### 2.3.1 Third-order Actuator Model With $\theta$ , $\dot{\theta}$ , $i_a$ As State Variables

By defining a 3x1 state vector of the  $i$ th actuator to be :

$$\mathbf{X}_{Ai}(t) = \begin{bmatrix} \theta_i(t) , \dot{\theta}_i(t) , i_{ai}(t) \end{bmatrix}^T , \quad (2.20)$$

equations (2.16) and (2.17) can be rewritten in state variable form as follows :

$$\dot{\mathbf{X}}_{Ai}(t) = \mathbf{A}_{Ai} \mathbf{X}_{Ai}(t) + \mathbf{B}_{Ai} U_i(t) + \mathbf{F}_{Ai} T_i(t) , \quad (2.21)$$

where

$$\mathbf{A}_{Ai} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & \frac{-B_{vi}}{J_{mi}} & \frac{k_{ti}}{N_i J_{mi}} \\ 0 & \frac{-k_{vi}}{L_i} & \frac{-R_i}{L_i} \end{bmatrix} , \quad \mathbf{F}_{Ai} = \begin{bmatrix} 0 \\ -\frac{1}{N_i^2 J_{mi}} \\ 0 \end{bmatrix} \quad (2.22)$$

$$\mathbf{B}_{Ai} = \begin{bmatrix} 0 \\ 0 \\ 1/L_i \end{bmatrix} , \quad U_i(t) = v_i(t) .$$

$\mathbf{X}_{Ai}(t)$  : 3x1 state vector of the  $i$ th actuator

$U_i(t)$  : scalar input to the  $i$ th actuator



$T_i(t)$  : the load acting on the  $i$ th actuator due to the manipulator links (from equation 2.1).

$A_{Ai}$ ,  $B_{Ai}$  and  $F_{Ai}$  are the system, input and load distribution matrices for the  $i$ th actuator respectively, with appropriate dimensions.

For  $N$  actuators ( $N$  dof robot manipulator), the augmented dynamic equation of the actuators can be written in the following compact form :

$$\dot{X}_A(t) = A_A X_A(t) + B_A U(t) + F_A T(t) , \quad (2.23)$$

where

$$\begin{aligned} X_A(t) &= \left[ X_{A1}^T(t) , X_{A2}^T(t) , \dots , X_{AN}^T(t) \right]^T \\ U(t) &= \left[ U_1(t) , U_2(t) , \dots , U_N(t) \right]^T \\ T(t) &= \left[ T_1(t) , T_2(t) , \dots , T_N(t) \right]^T \\ A_A &= \text{diag} \left[ A_{A1} , A_{A2} , \dots , A_{AN} \right] \\ B_A &= \text{diag} \left[ B_{A1} , B_{A2} , \dots , B_{AN} \right] \\ F_A &= \text{diag} \left[ F_{A1} , F_{A2} , \dots , F_{AN} \right] \end{aligned} \quad (2.24)$$

and  $X_A(t)$  is a  $3Nx1$  vector,  $U(t)$  is an  $Nx1$  input vector, and  $T(t)$  is the torque due to the mechanical link (equation 2.1).

It can be observed from the structure of the input matrix  $B_{Ai}$  and the load distribution matrix  $F_{Ai}$  in equation (2.22), hence the structure of the augmented matrices  $B_A$  and  $F_A$  respectively, that the load torque  $T_i(t)$  or  $T(t)$  lies outside the range space of the input matrix.

### 2.3.2 Third-order Actuator Model With $\theta$ , $\dot{\theta}$ , $\ddot{\theta}$ As State Variables

From equation (2.19), the second form of the actuator dynamic model in state variable form can be obtained. Here the armature current is replaced by

joint acceleration as state variable. Let the state vector for the  $i$ th actuator be defined as follows :

$$X_{Bi}(t) = \begin{bmatrix} \theta_i(t) & \dot{\theta}_i(t) & \ddot{\theta}_i(t) \end{bmatrix}^T. \quad (2.25)$$

Then, the state equation of the actuator dynamic model can be rewritten in the following form :

$$\dot{X}_{Bi}(t) = A_{Bi} X_{Bi}(t) + B_{Bi} U_i(t) + F_{Bi} T_i(t) + W_B \dot{T}_i(t), \quad (2.26)$$

where

$$A_{Bi} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -\frac{k_{vi}k_{ti}+B_{vi}R_i}{J_{mi}L_i} & -\frac{B_{vi}L_i+J_{mi}R_i}{J_{mi}L_i} \end{bmatrix}, \quad B_{Bi} = \begin{bmatrix} 0 \\ 0 \\ \frac{k_{ti}}{J_{mi}L_iN_i} \end{bmatrix}, \quad (2.27)$$

$$F_{Bi} = \begin{bmatrix} 0 \\ 0 \\ -\frac{R_i}{N_i^2 J_{mi} L_i} \end{bmatrix}, \quad W_{Bi} = \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{N_i^2 J_{mi}} \end{bmatrix}, \quad U_i(t) = v_i(t).$$

$X_{Bi}(t)$  : 3x1 state vector of the  $i$ th actuator

$U_i(t)$  : scalar input to the  $i$ th actuator

$T_i(t)$  : the load torque acting on the  $i$ th actuator due to the manipulator itself (from equation 1 or 2)

$\dot{T}_i(t)$  : the time derivative of the load torque  $T_i(t)$  acting on the  $i$ th actuator,

and  $A_{Bi}$ ,  $B_{Bi}$ ,  $F_{Bi}$ , and  $W_{Bi}$  are the system, input, load distribution and rate of load distribution matrices with appropriate dimensions respectively, for the  $i$ th actuator.

For an  $N$  dof robot manipulator, the augmented dynamic equation of the actuators can be written in compact form as follows :

$$\dot{X}_B(t) = A_B X_B(t) + B_B U(t) + F_B T(t) + W_B \dot{T}(t), \quad (2.28)$$

where

$$\begin{aligned} X_B(t) &= \left[ X_{B1}^T(t), X_{B2}^T(t), \dots, X_{BN}^T(t) \right]^T \\ U(t) &= \left[ U_1(t), U_2(t), \dots, U_N(t) \right]^T \\ T(t) &= \left[ T_1(t), T_2(t), \dots, T_N(t) \right]^T \\ \dot{T}(t) &= \left[ \dot{T}_1(t), \dot{T}_2(t), \dots, \dot{T}_N(t) \right]^T \\ A_B &= \text{diag} [ A_{B1}, A_{B2}, \dots, A_{BN} ] \\ B_B &= \text{diag} [ B_{B1}, B_{B2}, \dots, B_{BN} ] \\ F_B &= \text{diag} [ F_{B1}, F_{B2}, \dots, F_{BN} ] \\ W_B &= \text{diag} [ W_{B1}, W_{B2}, \dots, W_{BN} ] \end{aligned} \quad (2.29)$$

and  $X_B(t)$  is a  $3N \times 1$  vector,  $U(t)$  is an  $N \times 1$  input vector,  $T(t)$  is the  $N \times 1$  mechanical link torque (equation 2.1), and  $\dot{T}(t)$  is its time derivative.

It can be observed from the structure of the input matrix  $B_{Bi}$ , the load distribution matrix  $F_{Bi}$  and the derivative of the load distribution matrix  $W_{Bi}$  in equation (2.22) (hence the structure of the matrices  $B_B$ ,  $F_B$ , and  $W_B$ , in equation 2.29 respectively), that the load torque  $T_i(t)$  and its derivative  $\dot{T}_i(t)$  (hence the

vector  $T(t)$  and  $\dot{T}(t)$ , respectively) lie within the range space of the input matrix. Hence the control input which enters the system through the input matrix  $B_{B_i}$  or  $B_B$  can affect and compensate directly the nonlinear, coupled and time varying effect due to the load, that is, the mechanical linkage.

## 2.4 INTEGRATED DYNAMIC MODEL OF ROBOT MANIPULATOR

Eventhough the actuator dynamics constitute an important part of the complete robot manipulator system, only a limited number of research on deriving the integrated mathematical model comprising the mechanical part of the system and the actuators, as well as on the controller synthesis based on the complete robot manipulator model can be found in the literature. The explicit form of the integrated model of the robot manipulator can be obtained by combining the actuator dynamic equations with the dynamic equations of the mechanical links. Vukobratovic and Potkonjak [1982] derived the complete model of the robot manipulator based on the mechanical links and actuator models described by equations (2.1) and (2.23) respectively. The derivations are unnecessarily complicated. A much simpler formulation of the robot manipulator integrated model is outlined by Vukobratovic et.al. [1985], Troch [1986], Troch et.al. [1986]. All of the above methods are based on the actuators and mechanical links models represented by equations (2.23) and (2.1) respectively. All of these methods will give the complete robot manipulator dynamic model in state space form with the joint angles, velocities and motor armature currents as the state variables. For comparison purposes, their formulations are briefly outlined in Appendix C. Spong and Vidyasagar [1989] derived an integrated dynamic model of the robot manipulator based on a second-order actuator model (equation 2.18) for each joint. Thus, their overall mathematical model is greatly simplified. Tarn *et.al.* [1988, 1991] derived an integrated robot manipulator dynamic model based on a third-order actuator model for each joint represented by equation (2.19), and the mechanical link model described by equation (2.1). The joint angles, velocities and accelerations have been chosen as the state variables in their formulation. However, the actuators viscous friction terms have been assumed negligible, and

In this section, two different state space representations of the complete robot manipulator dynamic model are presented. The formulation of the first form of the integrated model is based on the actuator dynamic model described by equation (2.23), while the derivation of the second form of the integrated model of the manipulator and its actuating mechanism is based on the actuator dynamic model described by equation (2.28).

This method is based on the dynamic equation of the manipulator in state variable form of equation (2.5) and the actuators dynamic behaviour described by equation (2.23). Let the transformation between the manipulator state vector  $X_p(t)$  and the actuator state vector  $X_A(t)$  be  $\mathbb{Z}_A$ , such that,

where the  $2N \times 3N$  transformation matrix  $\mathbb{Z}_A$  has the following form :

71

Substitution of equation (2.30) into (2.5), gives

$$\mathbb{Z}_A \dot{X}_A(t) = A_p(X_A, \xi, t) \mathbb{Z}_A X_A(t) + B_p(X_A, \xi, t) T. \quad (2.32)$$

From equation (2.32), the driving forces/torques  $T$  can be obtained as

$$T = B_p^\dagger(X_A, \xi, t) \mathbb{Z}_A \dot{X}_A(t) - B_p^\dagger(X_A, \xi, t) A_p(X_A, \xi, t) \mathbb{Z}_A X_A(t), \quad (2.33)$$

where  $B_p^\dagger(X_A, \xi, t)$  is the Penrose-Pseudoinverse of  $B_p(X_A, \xi, t)$  :

$$B_p^\dagger(X_A, \xi, t) = \left[ B_p^T(X_A, \xi, t) B_p(X_A, \xi, t) \right]^{-1} B_p^T(X_A, \xi, t). \quad (2.34)$$

Substituting equation (2.33) into the actuators state equation (2.23), gives the state equation of the integrated robot manipulator model as :

$$\dot{X}_A(t) = A_A(X_A, \xi, t) X_A(t) + B_A(X_A, \xi, t) U(t), \quad (2.35)$$

where

$$A_A(X_A, \xi, t) = \left[ I_N - F_A B_p^\dagger(X_A, \xi, t) \mathbb{Z}_A \right]^{-1} \left[ A_A - F_A B_p^\dagger(X_A, \xi, t) A_p(X_A, \xi, t) \mathbb{Z}_A \right] \quad (2.36)$$

$$B_A(X_A, \xi, t) = \left[ I_N - F_A B_p^\dagger(X_A, \xi, t) \mathbb{Z}_A \right]^{-1} B_A. \quad (2.37)$$

In this method, it is required to find the pseudoinverse of the matrix  $B_p(X_A, \xi, t)$ . In the following, the existence and the uniqueness\* of the matrix  $B_p^\dagger(X_A, \xi, t)$  will be shown.

Since the manipulator inertia matrix  $M(X_A, \xi, t)$  is always symmetric and nonsingular, the following properties of the inertia matrix hold for any value of  $X_A(t)$ ,  $\xi$ , and  $t$  :

$$M(X_A, \xi, t) = M^T(X_A, \xi, t) \quad (2.38)$$

---

\* It should be noted that in general, the Penrose-Pseudoinverse of a matrix is not unique.

$$\left[ M^{-1}(X_A, \xi, t) \right]^T = M^{-1}(X_A, \xi, t) .$$

Hence from (2.9),

$$B_p^T(X_A, \xi, t) = \begin{bmatrix} O_{NN} & \vdots & M^{-1}(X_A, \xi, t) \end{bmatrix} . \quad (2.39)$$

This gives

$$B_p^T(X_A, \xi, t) B_p(X_A, \xi, t) = \begin{bmatrix} M^{-1}(X_A, \xi, t) \end{bmatrix}^2 , \quad (2.40)$$

and

$$\left[ B_p^T(X_A, \xi, t) B_p(X_A, \xi, t) \right]^{-1} = \begin{bmatrix} M(X_A, \xi, t) \end{bmatrix}^2 . \quad (2.41)$$

Thus, 
$$B_p^\dagger(X_A, \xi, t) = \begin{bmatrix} M(X_A, \xi, t) \end{bmatrix}^2 \begin{bmatrix} O_{NN} & \vdots & M^{-1}(X_A, \xi, t) \end{bmatrix}$$

$$= \begin{bmatrix} O_{NN} & \vdots & M(X_A, \xi, t) \end{bmatrix} . \quad (2.42)$$

Since  $M(X_A, \xi, t)$  exists and is unique, therefore,  $B_p^\dagger(X_A, \xi, t)$  also exists and is unique. This concludes the proof.  $\square$

Equation (2.42) not only provides the proof for the existence and the uniqueness of the matrix  $B_p^\dagger(X_A, \xi, t)$ , but also provides a simple method of determining the matrix  $B_p^\dagger(X_A, \xi, t)$ .

Due to the structures of the  $F_{A_i}$  and  $B_{A_i}$  matrices, hence, the structure of the  $F_A$  and  $B_A$  matrices, it is observed that equation (2.37) is equivalent to the actuators input matrix  $B_A$ , which is constant and independent of  $X_A(t)$ ,  $\xi$ , and  $t$  :

$$B_A(X_A, \xi, t) = B_A . \quad (2.43)$$

This can be verified from the structure of the actuator dynamic equations (2.16) and (2.17). Equations (2.16) and (2.17) are 'independent' of the input voltage  $v_i(t)$  and the load torque  $T_i(t)$ , respectively. Due to the structure of the load distribution matrix  $F_{A_i}$ , the components of the load torque  $T_i(t)$  (that is, the link inertias, the Coriolis and centrifugal forces, etc.) will coupled with the elements of the second row of the system matrix  $A_{B_i}$  only. Thus, equation (2.17)

remains the same when the  $i$ th mechanical link dynamic equation  $T_i(t)$ , which can be obtained from the mechanical link equation (2.1), is directly substituted into equation (2.16). Hence, the input term  $v_i(t)/L_i$  is unchanged for the integrated model for the  $i$ th link. Hence, the input matrix  $B_A(X_A, \xi, t)$  for the integrated model remains the same as the input matrix of the augmented actuator model  $B_A$ .

The method presented above is different from those outlined by Vukobratovic *et.al* [1985], Troch [1986], and Troch *et.al* [1986]. The main difference lies in the choice of the form of the dynamic equation for the mechanical linkage used in the formulation. Here, the formulation of the integrated model is based on the mechanical link dynamic model in state space form (2.5), while those in the references based their formulation on equation (2.1). Furthermore, the structure of the integrated dynamic model obtained here is slightly different from that of Vukobratovic *et.al* [1985], Troch [1986], and Troch *et.al* [1986]. However, as it is shown in the Appendix C, the integrated model derived above (equations 2.35, 2.36, and 2.37) is equivalent to those obtained by Vukobratovic *et.al* [1985], Troch [1986], and Troch *et.al* [1986].

#### 2.4.2 Integrated Model Based On $\theta, \dot{\theta}, \ddot{\theta}$ As State Variables

In this subsection, the integrated robotic model based on equation (2.28) of the actuator dynamics is presented. The derivation of the integrated model is not as straightforward as the previous one due to the need to find the time derivative of the dynamic equation of the mechanical part (mechanical link) of the manipulator.

From equation (2.1), the derivative of the torque,  $T(t)$ , may be written as :

$$\dot{T}(t) = M(\theta(t), \xi) \ddot{\theta}(t) + \tilde{C}(\theta(t), \dot{\theta}(t), \xi) \ddot{\theta}(t) + \tilde{D}(\theta(t), \dot{\theta}(t), \xi) \dot{\theta}(t) \quad (2.44)$$

where

$$\tilde{C}(\theta(t), \dot{\theta}(t), \xi) \ddot{\theta}(t) = \dot{M}(\theta(t), \xi) \ddot{\theta}(t) + \bar{D}(\theta(t), \xi) \dot{V}(\dot{\theta}(t)) \quad (2.45)$$

$$\tilde{D}(\theta(t), \dot{\theta}(t), \xi) \dot{\theta}(t) = \dot{\bar{D}}(\theta(t), \xi) V(\dot{\theta}(t)) + \dot{G}(\theta(t), \xi) , \quad (2.46)$$



and  $\tilde{C}(\theta(t), \dot{\theta}(t), \xi)$  and  $\tilde{D}(\theta(t), \dot{\theta}(t), \xi)$  are  $N \times N$  matrices. Now define the following transformations :

$$\begin{aligned}\ddot{\theta}(t) &= \mathbb{Z}_B \mathbf{X}_B(t) \\ \dot{\theta}(t) &= \mathbb{Z}_{B1} \mathbf{X}_B(t) \\ \theta(t) &= \mathbb{Z}_{B2} \mathbf{X}_B(t)\end{aligned}\tag{2.47}$$

where  $\mathbb{Z}_B$ ,  $\mathbb{Z}_{B1}$ , and  $\mathbb{Z}_{B2}$  have the following form :

$$\begin{aligned}\mathbb{Z}_B &= \begin{bmatrix} 0 & 0 & 1 & \vdots & & 0 \\ \text{---} & \text{---} & \text{---} & \vdots & & \\ & & 0 & 0 & 1 & \\ & & \text{---} & \text{---} & \text{---} & \ddots \\ & 0 & & & & \vdots & \text{---} & \text{---} & \text{---} \\ & & & & & & & 0 & 0 & 1 \end{bmatrix} \\ \mathbb{Z}_{B1} &= \begin{bmatrix} 0 & 1 & 0 & \vdots & & 0 \\ \text{---} & \text{---} & \text{---} & \vdots & & \\ & & 0 & 1 & 0 & \\ & & \text{---} & \text{---} & \text{---} & \ddots \\ & 0 & & & & \vdots & \text{---} & \text{---} & \text{---} \\ & & & & & & & 0 & 1 & 0 \end{bmatrix} \\ \mathbb{Z}_{B2} &= \begin{bmatrix} 1 & 0 & 0 & \vdots & & 0 \\ \text{---} & \text{---} & \text{---} & \vdots & & \\ & & 1 & 0 & 0 & \\ & & \text{---} & \text{---} & \text{---} & \ddots \\ & 0 & & & & \vdots & \text{---} & \text{---} & \text{---} \\ & & & & & & & 1 & 0 & 0 \end{bmatrix} .\end{aligned}\tag{2.48}$$

Equation (2.1) can be rewritten as :

$$T(t) = M(\theta(t), \xi) \ddot{\theta}(t) + \hat{D}(\theta(t), \dot{\theta}(t), \xi) \dot{\theta}(t) + \hat{G}(\theta(t), \xi) \theta(t) , \quad (2.49)$$

where  $\hat{D}(\theta(t), \dot{\theta}(t), \xi)$  and  $\hat{G}(\theta(t), \xi)$  are  $N \times N$  matrices. By substituting equations (2.44) and (2.49) into the augmented actuator dynamic equation (2.28), and using (2.47), the integrated dynamic model of the robotic system can be obtained as follows :

$$\dot{X}_B(t) = A_B(X_B, \xi, t) X_B(t) + B_B(X_B, \xi, t) U(t) , \quad (2.50)$$

where

$$\begin{aligned} A_B(X_B, \xi, t) = [ I_{3N} - W_B M(X_B, \xi, t) Z_B ]^{-1} \{ & A_B + [ F_B M(X_B, \xi, t) + \\ & W_B \tilde{C}(X_B, \xi, t) ] Z_B + [ F_B \hat{D}(X_B, \xi, t) + \\ & W_B \tilde{D}(X_B, \xi, t) ] Z_{B1} + F_B \hat{G}(X_B, \xi, t) Z_{B2} \} \end{aligned} \quad (2.51)$$

$$B_B(X_B, \xi, t) = [ I_{3N} - W_B M(X_B, \xi, t) Z_B ]^{-1} B_B . \quad (2.52)$$

Due to the structure of the load distribution matrix  $F_{Bi}$  and the derivative of the load distribution matrix  $W_{Bi}$  in equation (2.27), the nonlinear components of the load torque  $T_i(t)$  and its derivative  $\dot{T}_i(t)$ ; that is the link inertias, the Coriolis and Centrifugal forces and the gravitational forces, and their corresponding derivatives; will be integrated into the last row of the system matrix  $A_{Bi}$ . Hence, in the resultant integrated model, the nonlinear, coupled and uncertain components of the load torque  $T(t)$  and its derivative will appear in very 3 $i$ th ( $i=1, 2, \dots, N$ ) row of the system matrix  $A_B(X_B, \xi, t)$  and input matrix  $B_B(X_B, \xi, t)$ . Thus, in this case, the elements of the input matrix  $B_B(X_B, \xi, t)$  are highly nonlinear, coupled, time varying and contain uncertainties.

This method will be applied to a three dof robot manipulator in section 2.5 to derive the complete equation of motion for the system.

### 2.4.3 Differences Between The Two Integrated Models

For convenience, the integrated model with the  $\theta(t)$ ,  $\dot{\theta}(t)$ ,  $i_a(t)$  as the state variables, described by equations (2.35), (2.36) and (2.37), will be referred to as Form A, and the integrated model with the  $\theta(t)$ ,  $\dot{\theta}(t)$ ,  $\ddot{\theta}(t)$  as the state variables, represented by equations (2.50), (2.51) and (2.52), will be referred to as Form B.

The formulation of the overall integrated model of the manipulator and the actuator dynamics based on Form A is simple and straightforward. The input matrix  $B_A(X_A, t)$  is equivalent to the input matrix of the augmented actuators state equation, thus simplifying the derivation of the integrated dynamic equation. However, the formulation results in the nonlinear, uncertain and coupling terms to lie outside the range space of the input matrix of the integrated model state equation. Thus, even if the nonlinearities and uncertainties are known, one cannot compensate them since the control input which enters through the input matrix cannot affect the nonlinearities and the uncertainties. Hence, it is difficult to design an advanced and robust controller for the robot manipulator when the overall dynamic model is in the Form A.

The structure of the integrated model based on Form B is different from that of Form A in that the nonlinear, uncertain and coupling terms lie within the range space of the input matrix of the derived overall state equation. Thus, the control input which enters the system through the input matrix can affect and compensate directly for the nonlinear and uncertain components. However, the derivation is not as straightforward as in Form A, because it is necessary to find the time derivative of the nonlinear, coupled dynamic equation of the mechanical part of the manipulator, which is very time consuming and tedious especially for a robot manipulator with a large dof. Nevertheless, once the integrated model is obtained in the form B, various advanced and robust control strategies can be developed for controlling the nonlinear robot manipulator system. In view of this advantage over Form A, the formulation of the complete robot manipulator mathematical model as presented in section 2.4.2 is used in this study.

## 2.5 DYNAMIC MODEL OF A THREE dof REVOLUTE ROBOT MANIPULATOR

In this section, a complete mathematical model of a three dof revolute robot manipulator, as shown in Figure 2.3 is derived based on the approach presented in section 2.4.2. The robot manipulator is actuated by three permanent magnet armature controlled DC motors. The parameters of the mechanical linkage and the actuators as presented in Zainol Anuar [1985] are tabulated in Table 2-1 and Table 2-2 respectively. It is assumed that the permanent magnet DC motors are identical for all the three joints. It is also assumed that the end-effector and the variable load are lumped together as a single mass  $m_L$  and is located at the end of the third link. This load is assumed to vary between 0 kg to 20 kg.

### 2.5.1 Dynamic Equation Of The Mechanical Linkage

From the Euler-Lagrange method outlined in Appendix B, the dynamic equations of for the three dof robot manipulator under consideration can be written in the following form :

$$M(\theta, \xi) \ddot{\theta}(t) + \bar{D}(\theta, \xi) \hat{V}(\dot{\theta}) + G(\theta, \xi) = T(t) , \quad (2.53)$$

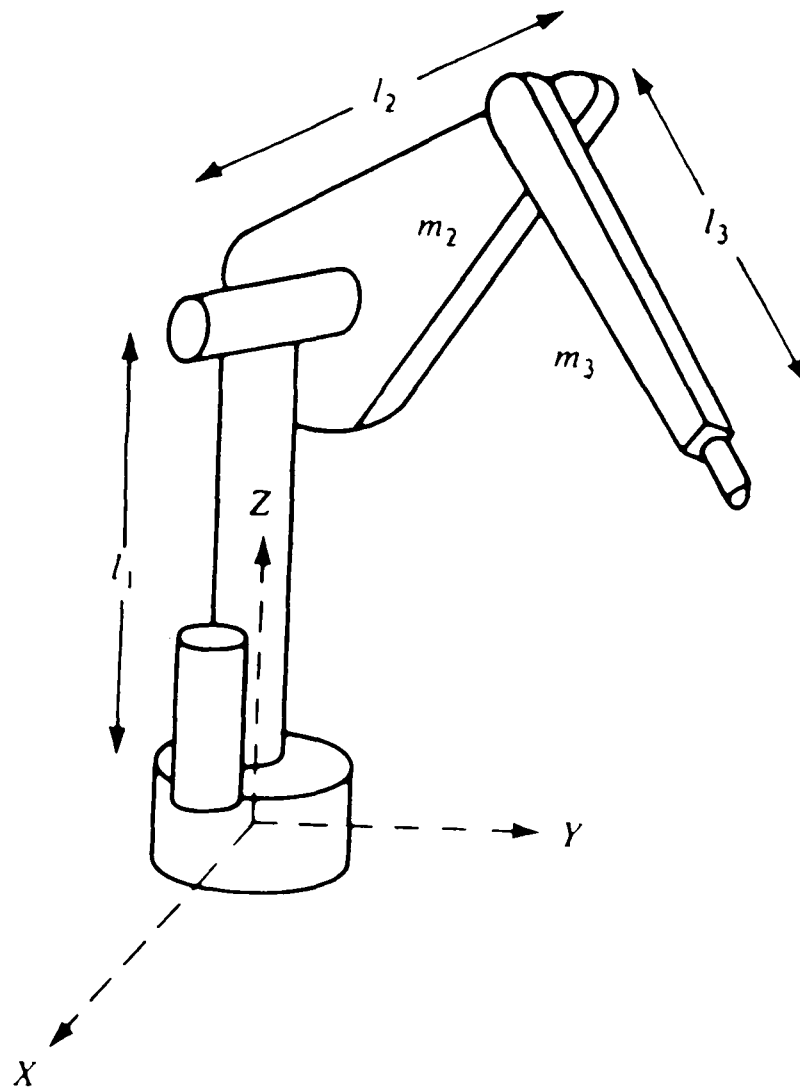
where

$$\theta(t) = [ \theta_1(t) \quad \theta_2(t) \quad \theta_3(t) ]^T , \quad (2.54)$$

$$T(t) = [ T_1(t) \quad T_2(t) \quad T_3(t) ]^T , \quad (2.55)$$

$$\xi = m_L \triangleq \text{the payload mass carried by the manipulator}$$

$$M(\theta, \xi) = \begin{bmatrix} M_{11} & 0 & 0 \\ 0 & M_{22} & M_{23} \\ 0 & M_{23} & M_{33} \end{bmatrix} , \quad (2.56)$$



Manipulator's Range of Operation :

Joint 1 :  $-160^\circ \leq \theta_1 \leq 160^\circ$  ;  $\dot{\theta}_{1max} = 120^\circ/\text{s}$

Joint 2 :  $-225^\circ \leq \theta_2 \leq 45^\circ$  ;  $\dot{\theta}_{2max} = 110^\circ/\text{s}$

Joint 3 :  $-46^\circ \leq \theta_3 \leq 225^\circ$  ;  $\dot{\theta}_{3max} = 110^\circ/\text{s}$

FIGURE 2.3 : Three dof Revolute Robot Manipulator

	Link Number			
	0	1	2	3
Mass, $m_i$ (kg)			2.0	1.5
Length, $l_i$ (m)	0.3	0.5	0.6	0.5
Position of the center of gravity (m)			0.3	0.25
Moment of inertia $I_i$ , ( $\text{kgm}^2$ ) with respect to :				
x-axis at $\text{cg}^*$			0.06	0.03
y-axis at $\text{cg}^*$			0.008	0.0025
z-axis at $\text{cg}^*$		0.04	0.06	0.03
$\text{cg}^*$ : centre of gravity				

**TABLE 2-1 : Parameters Of The Three dof Robot Manipulator Linkage**

Moment of inertia, $J_{mi}$	:	1.52 kgm <sup>2</sup>
Armature resistance, $R_i$	:	2.45 $\Omega$
Armature inductance, $L_i$	:	0.245 H
Viscous friction constant, $B_{vi}$	:	1.5 Nm/rad/s
Back Emf constant, $k_{vi}$	:	7.0 V/rad/s
Motor torque constant, $k_{ti}$	:	4.3 Nm/A
Inverse of gear ratio :		
joint 1 , $N_1$	:	16
joint 2 , $N_2$	:	18
joint 3 , $N_3$	:	18
$i = 1, 2, 3 .$		

**TABLE 2-2 : Parameters Of The Actuators**

$$\bar{D}(\theta, \xi) = \begin{bmatrix} 0 & \bar{D}_{12} & \bar{D}_{13} & 0 & 0 & 0 \\ \bar{D}_{21} & 0 & 0 & 0 & \bar{D}_{25} & \bar{D}_{26} \\ \bar{D}_{31} & 0 & 0 & \bar{D}_{34} & 0 & 0 \end{bmatrix}, \quad (2.57)$$

$$\hat{V}(\dot{\theta}) = \begin{bmatrix} \dot{\theta}_1 \dot{\theta}_1 \\ \dot{\theta}_1 \dot{\theta}_2 \\ \dot{\theta}_1 \dot{\theta}_3 \\ \dot{\theta}_2 \dot{\theta}_2 \\ \dot{\theta}_2 \dot{\theta}_3 \\ \dot{\theta}_3 \dot{\theta}_3 \end{bmatrix}, \quad \text{and} \quad G(\theta, \xi) = \begin{bmatrix} 0 \\ G_2 \\ G_3 \end{bmatrix}, \quad (2.58)$$

where the non-zero elements of the matrices  $M(\theta, \xi)$ ,  $\bar{D}(\theta, \xi)$  and  $G(\theta, \xi)$  are as follows :

$$M_{11} = \mathcal{A}_1 + \mathcal{A}_2 \cos^2(\theta_2) + \mathcal{A}_3 \cos^2(\theta_2 + \theta_3) + \mathcal{A}_4 \cos(\theta_2) \cos(\theta_2 + \theta_3)$$

$$M_{22} = \mathcal{A}_5 + \mathcal{A}_4 \cos(\theta_3)$$

$$M_{23} = \mathcal{A}_6 + \mathcal{A}_7 \cos(\theta_3)$$

$$M_{33} = \mathcal{A}_8$$

$$\bar{D}_{12} = -\mathcal{A}_2 \sin(2\theta_2) - \mathcal{A}_3 \sin(2\theta_2 + 2\theta_3) - \mathcal{A}_4 \sin(2\theta_2 + \theta_3)$$

$$\bar{D}_{13} = -\mathcal{A}_3 \sin(2\theta_2 + 2\theta_3) - \mathcal{A}_4 \cos(\theta_2) \sin(2\theta_2 + \theta_3)$$



$$\begin{aligned}
\bar{D}_{21} &= -\frac{\bar{D}_{12}}{2} = 0.5\mathcal{A}_2 \sin(2\theta_2) + 0.5\mathcal{A}_3 \sin(2\theta_2+2\theta_3) + 0.5\mathcal{A}_4 \sin(2\theta_2+\theta_3) \\
\bar{D}_{25} &= -\mathcal{A}_4 \sin(\theta_3) \\
\bar{D}_{26} &= \frac{\bar{D}_{25}}{2} = -0.5\mathcal{A}_4 \sin(\theta_3) \\
\bar{D}_{31} &= -\frac{\bar{D}_{13}}{2} = 0.5\mathcal{A}_3 \sin(2\theta_2+2\theta_3) + 0.5\mathcal{A}_4 \cos(\theta_2) \sin(2\theta_2+\theta_3) \\
\bar{D}_{34} &= -\bar{D}_{26} = 0.5\mathcal{A}_4 \sin(\theta_3) \\
G_2 &= \mathcal{A}_9 \cos(\theta_2) + \mathcal{A}_{10} \cos(\theta_2+\theta_3) \\
G_3 &= \mathcal{A}_{10} \cos(\theta_2+\theta_3) ,
\end{aligned} \tag{2.59}$$

where

$$\begin{aligned}
\mathcal{A}_1 &= \mathbb{I}_{33}^1 + \mathbb{I}_{22}^2 + \mathbb{I}_{22}^3 \\
\mathcal{A}_2 &= [0.25 m_2 + m_3 + m_L] \ell_2^2 + \mathbb{I}_{11}^2 - \mathbb{I}_{22}^2 \\
\mathcal{A}_3 &= 0.25 [m_3 + 4 m_L] \ell_3^2 + \mathbb{I}_{11}^3 - \mathbb{I}_{22}^3 \\
\mathcal{A}_4 &= [m_3 + 2 m_L] l_2 l_3 \\
\mathcal{A}_5 &= [0.25 m_2 + m_3 + m_L] \ell_2^2 + 0.25 [m_3 + 4 m_L] \ell_3^2 + \mathbb{I}_{11}^2 - \mathbb{I}_{11}^3 \\
\mathcal{A}_6 &= 0.25 [m_3 + 4 m_L] \ell_3^2 + \mathbb{I}_{11}^3 \\
\mathcal{A}_7 &= 0.5 \mathcal{A}_4 = 0.5 [m_3 + 2 m_L] l_2 l_3 \\
\mathcal{A}_8 &= 0.25 [m_3 + 4 m_L] \ell_3^2 + \mathbb{I}_{11}^3 \\
\mathcal{A}_9 &= -[0.5 m_2 + m_3 + m_L] g l_2
\end{aligned} \tag{2.60}$$

$$\mathcal{A}_{10} = -[0.5 m_3 + m_L] g l_3 ,$$

where  $g=9.8016 \text{ m/s}^2$ ,  $\mathbb{I}_{11}^i$ ,  $\mathbb{I}_{22}^i$ , and  $\mathbb{I}_{33}^i$  are the moment of inertia at the centre of gravity of the  $i$ th link with respect to the x-axis, y-axis and z-axis respectively.

The terms  $\bar{D}(\theta, \xi)\hat{V}(\dot{\theta})$  and  $G(\theta, \xi)$  in equation (2.53) can be rewritten as  $\hat{D}(\theta, \dot{\theta}, \xi)\dot{\theta}$  and  $\hat{G}(\theta, \xi)\theta$ , respectively. Thus, the manipulator dynamics can be rewritten as

$$T(t) = M(\theta, \xi) \ddot{\theta}(t) + \hat{D}(\theta, \dot{\theta}, \xi) \dot{\theta} + \hat{G}(\theta, \xi) \theta , \quad (2.61)$$

where

$$\hat{D}(\theta, \dot{\theta}, \xi) = \begin{bmatrix} 0 & \bar{D}_{12}\dot{\theta}_1 & \bar{D}_{13}\dot{\theta}_1 \\ \bar{D}_{21}\dot{\theta}_1 & 0 & \bar{D}_{25}\dot{\theta}_2 + \bar{D}_{26}\dot{\theta}_3 \\ \bar{D}_{31}\dot{\theta}_1 & \bar{D}_{34}\dot{\theta}_2 & 0 \end{bmatrix} , \quad (2.62)$$

and

$$\hat{G}(\theta, \xi) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{G_2}{\theta_2} & 0 \\ 0 & 0 & \frac{G_3}{\theta_3} \end{bmatrix} . \quad (2.63)$$

In order to apply the method presented in section 2.4.2, the dynamic equation of the robot manipulator, that is equation 2.53, is needed to be differentiated with respect to time, as presented next.

### 2.5.2 Derivative Of The Mechanical Link Torque

From equations (2.44), (2.45), and (2.46), the derivative of the torque (equation 2.53), with respect to time, for the three dof robot manipulator can be obtained as follows :

$$\dot{T}(t) = M(\theta, \xi) \ddot{\theta} + \tilde{C}(\theta, \dot{\theta}, \xi) \ddot{\theta} + \tilde{D}(\theta, \dot{\theta}, \xi) \dot{\theta} \quad (2.64)$$

where

$$\tilde{C}(\theta, \dot{\theta}, \xi) = \begin{bmatrix} \tilde{C}_{11} & \tilde{C}_{12} & \tilde{C}_{13} \\ \tilde{C}_{21} & \tilde{C}_{22} & \tilde{C}_{23} \\ \tilde{C}_{31} & \tilde{C}_{32} & 0 \end{bmatrix}, \quad (2.65)$$

$$\tilde{D}(\theta, \dot{\theta}, \xi) = \begin{bmatrix} 0 & \tilde{D}_{12} & \tilde{D}_{13} \\ \tilde{D}_{21} & \tilde{D}_{22} & \tilde{D}_{23} \\ \tilde{D}_{31} & \tilde{D}_{32} & \tilde{D}_{33} \end{bmatrix}, \quad (2.66)$$

and the elements of the  $\tilde{C}(\theta, \dot{\theta}, \xi)$  and  $\tilde{D}(\theta, \dot{\theta}, \xi)$  matrices are as follows :

$$\tilde{C}_{11} = 2 [ \bar{D}_{12} \dot{\theta}_2 + \bar{D}_{13} \dot{\theta}_3 ]$$

$$\tilde{C}_{12} = \bar{D}_{12} \dot{\theta}_1$$

$$\tilde{C}_{13} = \bar{D}_{13} \dot{\theta}_1$$

$$\tilde{C}_{21} = 2 \bar{D}_{21} \dot{\theta}_1$$

$$\tilde{C}_{22} = 2 \bar{D}_{25} \dot{\theta}_3$$

$$\tilde{C}_{23} = \bar{D}_{25} (\dot{\theta}_2 + \dot{\theta}_3) + \bar{D}_{26} \dot{\theta}_3$$

$$\tilde{C}_{31} = 2 \bar{D}_{31} \dot{\theta}_1$$

$$\tilde{C}_{32} = 2 \bar{D}_{34} \dot{\theta}_2 + \bar{D}_{26} \dot{\theta}_3$$

$$\begin{aligned} \tilde{D}_{12} = & -2 \mathcal{A}_2 \dot{\theta}_1 \dot{\theta}_2 \text{Cos}(2\theta_2) - 2 \mathcal{A}_3 \dot{\theta}_1 (\dot{\theta}_2 + \dot{\theta}_3) \text{Cos}(2\dot{\theta}_2 + 2\dot{\theta}_3) - \\ & \mathcal{A}_4 \dot{\theta}_1 (2\dot{\theta}_2 + \dot{\theta}_3) \text{Cos}(2\dot{\theta}_2 + \dot{\theta}_3) \end{aligned} \quad (2.67)$$

$$\begin{aligned} \tilde{D}_{13} = & -2 \mathcal{A}_3 \dot{\theta}_1 (\dot{\theta}_2 + \dot{\theta}_3) \text{Cos}(2\dot{\theta}_2 + 2\dot{\theta}_3) - \mathcal{A}_4 \dot{\theta}_1 \dot{\theta}_2 \text{Cos}(2\dot{\theta}_2 + \dot{\theta}_3) - \\ & \mathcal{A}_4 \dot{\theta}_1 \dot{\theta}_3 \text{Cos}(\theta_2) \text{Cos}(\dot{\theta}_2 + \dot{\theta}_3) \end{aligned}$$

$$\tilde{D}_{21} = -0.5 \tilde{D}_{12}$$

$$\tilde{D}_{22} = -\mathcal{A}_9 \text{Sin}(\theta_2) - \mathcal{A}_{10} \text{Sin}(\theta_2 + \theta_3)$$

$$\tilde{D}_{23} = -\mathcal{A}_{10} \text{Sin}(\theta_2 + \theta_3) - \mathcal{A}_4 \dot{\theta}_3 (\dot{\theta}_2 + 0.5 \dot{\theta}_3) \text{Cos}(\theta_3)$$

$$\tilde{D}_{31} = -0.5 \tilde{D}_{13}$$

$$\tilde{D}_{32} = -\mathcal{A}_{10} \text{Sin}(\theta_2 + \theta_3) + 0.5 \mathcal{A}_4 \dot{\theta}_2 \dot{\theta}_3 \text{Cos}(\theta_3)$$

$$\tilde{D}_{33} = -\mathcal{A}_{10} \text{Sin}(\theta_2 + \theta_3) .$$

Next, the dynamics of the actuating servo-mechanisms will be presented.

### 2.5.3 Model Of The DC Motors

From equation (2.28), the augmented actuators dynamic equation for the three dof robot manipulator is as follows :

$$\dot{X}_B(t) = A_B X_B(t) + B_B U(t) + F_B T(t) + W_B \dot{T}(t), \quad (2.68)$$

where

$$\begin{aligned} X_B(t) &= \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 \end{bmatrix}^T \\ &= \begin{bmatrix} x_{B1}^1 & x_{B1}^2 & x_{B1}^3 & x_{B2}^1 & x_{B2}^2 & x_{B2}^3 & x_{B3}^1 & x_{B3}^2 & x_{B3}^3 \end{bmatrix}^T \\ &= \begin{bmatrix} \theta_1 & \dot{\theta}_1 & \ddot{\theta}_1 & \theta_2 & \dot{\theta}_2 & \ddot{\theta}_2 & \theta_3 & \dot{\theta}_3 & \ddot{\theta}_3 \end{bmatrix}^T, \end{aligned} \quad (2.69)$$

$$A_B = \begin{bmatrix} 0 & 1 & 0 & & & & & & \\ 0 & 0 & 1 & & & & & & \\ 0 & a_{B32} & a_{B33} & & & & & & \\ \hline & & & 0 & 1 & 0 & & & \\ & & & 0 & 0 & 1 & & & \\ & & & 0 & a_{B65} & a_{B66} & & & \\ \hline & & & & & & 0 & 1 & 0 \\ & & & & & & 0 & 0 & 1 \\ & & & & & & 0 & a_{B98} & a_{B99} \end{bmatrix},$$

$$B_B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ b_{B1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & b_{B2} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & b_{B3} \end{bmatrix}, \quad F_B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ f_{B1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & f_{B2} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & f_{B3} \end{bmatrix}, \quad (2.70)$$

$$W_B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ w_{B1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & w_{B2} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & w_{B3} \end{bmatrix}, \quad U(t) = \begin{bmatrix} U_1(t) \\ U_2(t) \\ U_3(t) \end{bmatrix},$$

$$T(t) = \begin{bmatrix} T_1(t) \\ T_2(t) \\ T_3(t) \end{bmatrix}, \quad \dot{T}(t) = \begin{bmatrix} \dot{T}_1(t) \\ \dot{T}_2(t) \\ \dot{T}_3(t) \end{bmatrix}.$$

The non-zero elements of the  $A_B$  ,  $B_B$  ,  $F_B$  , and  $W_B$  matrices are as follow :

$$\begin{aligned}
a_{B32} &= - \frac{k_{v1}k_{t1}+B_{v1}R_1}{J_{m1}L_1} \quad , \quad a_{B33} = - \frac{B_{v1}L_1+J_{m1}R_1}{J_{m1}L_1} \\
a_{B65} &= - \frac{k_{v2}k_{t2}+B_{v2}R_2}{J_{m2}L_2} \quad , \quad a_{B66} = - \frac{B_{v2}L_2+J_{m2}R_2}{J_{m2}L_2} \\
a_{B98} &= - \frac{k_{v3}k_{t3}+B_{v3}R_3}{J_{m3}L_3} \quad , \quad a_{B99} = - \frac{B_{v3}L_3+J_{m3}R_3}{J_{m3}L_3} \\
b_{B1} &= \frac{k_{t1}}{J_{m1}L_1N_1} \quad , \quad b_{B2} = \frac{k_{t2}}{J_{m2}L_2N_2} \quad , \quad b_{B3} = \frac{k_{t3}}{J_{m3}L_3N_3} \\
f_{B1} &= - \frac{R_1}{N_1^2J_{m1}L_1} \quad , \quad f_{B2} = - \frac{R_2}{N_2^2J_{m2}L_2} \quad , \quad f_{B3} = - \frac{R_3}{N_3^2J_{m3}L_3} \\
w_{B1} &= - \frac{1}{N_1^2J_{m1}} \quad , \quad w_{B2} = - \frac{1}{N_2^2J_{m2}} \quad , \quad w_{B3} = - \frac{1}{N_3^2J_{m3}} .
\end{aligned} \tag{2.71}$$

#### 2.5.4 Integrated Model Of The Three dof Robot Manipulator

The complete dynamic model of the three dof robot manipulator can be obtained by using the formulae as presented in section 2.4.2. From equation (2.48), the 3x9 transformation matrices  $Z_B$ ,  $Z_{B1}$ , and  $Z_{B2}$  are as follows :

$$Z_B = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} , \tag{2.72}$$

$$\mathbb{Z}_{B1} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad (2.73)$$

$$\mathbb{Z}_{B2} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}. \quad (2.74)$$

Then, from equations (2.50), (2.51) and (2.52), the integrated model of the three dof robot manipulator can be obtained and has the following form :

$$\dot{X}_B(t) = A(X_B, \xi, t) X_B(t) + B(X_B, \xi, t) U(t), \quad (2.75)$$

where

$$A(X_B, \xi, t) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{32} & a_{33} & 0 & a_{35} & a_{36} & 0 & a_{38} & a_{39} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} & a_{67} & a_{68} & a_{69} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_{92} & a_{93} & a_{94} & a_{95} & a_{96} & 0 & a_{98} & a_{99} \end{bmatrix}, \quad (2.76)$$



$$B(X_B, \xi, t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ b_{11} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & b_{22} & b_{23} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & b_{32} & b_{33} \end{bmatrix}. \quad (2.77)$$

The non-zero nonlinear elements of the matrices  $A(X_B, \xi, t)$  and  $B(X_B, \xi, t)$  are given in Appendix D.

It can be observed from Appendix D and the above equations that the complete dynamic equations of the robot manipulator are highly nonlinear, coupled, and time varying. Furthermore, a varying payload carried by the manipulator during a task cycle will create uncertainties in the manipulator dynamics. Each non-zero element of the system and input matrices is a nonlinear function of the instantaneous configuration of the manipulator linkage, its instantaneous velocity, and payloads.

It can be observed from equations (2.76) and (2.77) that the nonlinear elements of the system matrix  $A(X_B, \xi, t)$  lie within the range space of the input matrix  $B(X_B, \xi, t)$ .

## 2.6 CONCLUSION

Two methods of deriving a more realistic dynamic mathematical model of a robot manipulator have been described in this chapter. The derived model of the integrated system comprises the mechanical part of the system as well as the

actuators and the gear trains. It is shown that the selection of an appropriate set of state variables for the robot manipulator system is vital for synthesizing an advanced and robust controller to overcome the nonlinearities, uncertainties and couplings inherent in the robot manipulator system. Eventhough deriving the complete model of the robot manipulator with the joint angles, velocities, and accelerations as the state variables is complex, it is postulated that the resulting mathematical model will lead to a convenient approach for the synthesis of advanced and robust control algorithm. Furthermore, the state variables are readily accessible for direct measurements using encoders, tachometers, and accelerometers. These measurements are usually very precise and are directly available for advanced control applications.

## CHAPTER 3

# ROBOT MANIPULATOR AS A LARGE SCALE UNCERTAIN SYSTEM

### 3.1 INTRODUCTION

In this research, large scale systems may be defined as those systems which are composed of a great number of diverse parts or those systems of high complexity. They are often also called interconnected systems or composite systems since the systems may be often be viewed as a collection of several interconnected subsystems.

Central to the control problem of a large scale system is the decomposition of the original system into a set of interconnected subsystems each of considerably lower order than the overall (original) system. The decomposition process of a large scale system into several lower order interconnected subsystems with appropriate form is by no means a trivial task. The process of decomposition of the global system must take into account properties like controllability and observability of the resulting subsystems so that the decoupled subsystems can be controlled and stabilized.

There are numerous ways of partitioning a large scale system into a set of interconnected subsystems as long as the above mentioned properties are satisfied. Two general approaches of decomposing a large scale system are [Michel and Miller, 1977] :

1. the structural properties of the system being modelled usually dictate a natural decomposition of the global system;
2. the decomposition is usually influenced by mathematical convenience to overcome technical difficulties.

Once the large scale system has been decomposed into a set of interconnected subsystems of manageable size, it becomes feasible to have these subsystems controlled by local controllers and possibly a global controller in such a way that the overall system performs as required in terms of stability, tracking, or other objectives. Such an approach leads to a decentralized and/or hierarchical control structure of the system.

Although robot manipulators are not a true large scale system in the sense that their dimensions are not as large as, for example, an electrical power network; due to the structural properties and complexity of the robot manipulators dynamics, robot manipulators may be considered as large scale systems. Thus, the control methodologies for large scale systems, such as decentralized and hierarchical control strategies, can be applied for controlling robot manipulators.

The aim of this chapter is to decompose and reduce the integrated robot manipulator dynamic model into a set of interconnected subsystems with bounded uncertainties, such that decentralized and hierarchical control concepts coupled with a deterministic approach can be applied to the tracking control problem of the robot manipulator. This chapter is organized as follows : in section 3.2, the process of decomposition of a robot manipulator into subsystems and couplings is presented; in section 3.3, based on the known allowable range of operation of the robot manipulator, the robot manipulator model is transformed into an interconnected subsystem description with bounded uncertainties; in section 3.4, the decomposition and transformation process is applied to the three dof revolute robot manipulator model as derived in Chapter 2.

## **3.2 DECOMPOSITION OF ROBOT MANIPULATOR**

In order to apply decentralized and/or hierarchical control method to robot manipulators, it is necessary to perform the distribution of the robot manipulators integrated dynamic model into local subsystems and couplings.

The decomposition of a robot manipulator depends mainly on the physical consideration of the system. As mentioned in the previous chapters, for an  $N$  dof robot manipulator, it consists of  $N$  rigid links connected in series by  $N$  joints. Each joint is powered by an actuator which, in turn, produces the motion of the joints. The dynamic behaviour of a joint influences, and is influenced by, the behaviour of the rest of the robot manipulators joints. Thus, it is common to treat each joint ( or each dof ) of the robot manipulator as a subsystem.

Based on the above observation, Vukobratovic *et.al.* [1980], Vukobratovic and Stokic [1983], Stokic and Vukobratovic [1984], Zainol Anuar [1985], Mills and Goldenberg [1988], and Pandian *et.al.* [1988], decomposed the robot manipulator into  $N$  interconnected subsystems each represented by the mathematical model of the actuator, and treated the mathematical model of the mechanical linkage of the robot manipulator represented by equation (1.1) or (2.1) as the interconnection function between the subsystems. The state vector and input variables of the actuating mechanism at the joint are taken as the state vector and the input variables for the subsystem, respectively. The decomposition results in completely linear subsystem models (actuator dynamics), and nonlinear interconnection functions (manipulator linkage's dynamics). However, this decomposition places part of the mechanical linkage's dynamics, which should be incorporated into the decoupled subsystem model description, into the interaction function. For example, the inertia at the  $i$ th joint due to the acceleration at the  $i$ th joint caused by the considered actuator should be included with the considered actuator's own inertia. Mathematically, since the interaction function (mechanical linkage's dynamics) is also a function of the state coordinates (state variables) of the actuating mechanism, those terms in the mechanical linkage's dynamic equation which are dependent on the state vector of the subsystem should be incorporated into the decoupled subsystem dynamic model. Another problem associated with this decomposition is that since the interaction function which is a differential equations on its own right, with the same order as the actuator dynamic equations, it will be difficult to design a tracking controller for the system, unless, the whole interaction function, that is, the mechanical linkage's dynamics, is treated as disturbances and the controller design is based only on the decoupled subsystems dynamics. As mentioned in previous chapters, this will result with poor robot manipulator performance for a fast robot

manipulator system.

Thus, in order to obtain a complete and realistic model of the robot manipulator described in the form of interconnected subsystems, the decomposition process should be based on the integrated model of the robot manipulator. That is, the integrated or complete model of the robot manipulator, which consists of the actuator dynamics and the dynamics of the mechanical part of the manipulator, should be obtained first, and then decomposed into an interconnected subsystem description. In the following subsections, two different descriptions of the robot manipulator in the input decentralized form are outlined. In both of these decompositions, each joint (each dof) of the robot manipulator is taken as a subsystem.

### 3.2.1 Decomposition Based On Integrated Model With Joint Angle, Velocity And Current As State Variables

In this subsection, the decomposition is based on the robot manipulator's integrated model in the form of equations (2.35), (2.36) and (2.37), as given in section 2.4.1 of Chapter 2.

Let the system matrix  $A_A(X_A, \xi, t)$  in equation (2.36) be partition into the following form :

$$A_A(X_A, \xi, t) = \begin{bmatrix} A_{A11}(X_A, \xi, t) & A_{A12}(X_A, \xi, t) & \dots & A_{A1N}(X_A, \xi, t) \\ - & - & - & - \\ A_{A21}(X_A, \xi, t) & A_{A22}(X_A, \xi, t) & \dots & A_{A2N}(X_A, \xi, t) \\ - & - & - & - \\ \vdots & \vdots & \ddots & \vdots \\ - & - & - & - \\ A_{AN1}(X_A, \xi, t) & A_{AN2}(X_A, \xi, t) & \dots & A_{ANN}(X_A, \xi, t) \end{bmatrix}, \quad (3.1)$$

where

$$A_{Aij}(X_A, \xi, t) \quad : \quad 3 \times 3 \text{ matrix}$$

$$i, j = 1, 2, \dots, N.$$

Then, since the input matrix  $B_A(X_A, \xi, t)$  in equation (2.37) is equivalent to the augmented actuators input matrix  $B_A$  which is in block diagonal form, the integrated robot manipulator dynamic equation (2.35) can be decomposed into a set of interconnected subsystems which is as follows :

$$\dot{X}_{Ai}(t) = A_{Aii}(X_A, \xi, t) X_{Ai}(t) + B_{Ai} U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{Aij}(X_A, \xi, t) X_{Aj}(t), \quad (3.2)$$

and

$$X_{Ai}(t) = \begin{bmatrix} \theta_i & \dot{\theta}_i & i_{ai} \end{bmatrix}^T, \quad (3.3)$$

$$X_{Ai}(t) \in \mathbb{R}^3, \quad i \in \mathcal{J},$$

where

$X_{Ai}(t)$  :  $i$ th component of the  $X_A(t)$  vector  $\equiv$  state vector of the  $i$ th actuator

$A_{Aij}(X_A, \xi, t)$  :  $3 \times 3$   $ij$ th submatrix of  $A_A(X_A, \xi, t)$

$B_{Ai}$  :  $3 \times 1$   $i$ th diagonal submatrix of  $B_A \equiv$  input matrix of the  $i$ th actuator

$U_i(t)$  :  $i$ th row of input vector  $U(t) \equiv$  input to the  $i$ th actuator

$X_{Aj}(t)$  :  $j$ th component of the  $X_A(t)$  state vector .

The structure of the  $A_{Aii}(X_A, \xi, t)$ ,  $B_{Ai}$ , and  $A_{Aij}(X_A, \xi, t)$  matrices may be obtained as follow :

$$A_{Aii}(X_A, \xi, t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & ** & ** \\ 0 & \frac{-k_{vi}}{L_i} & \frac{-R_i}{L_i} \end{bmatrix}, \quad B_{Ai} = \begin{bmatrix} 0 \\ 0 \\ 1/L_i \end{bmatrix}, \quad (3.4)$$

$$A_{Aij}(X_A, \xi, t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & ** & ** \\ 0 & 0 & 0 \end{bmatrix}, \quad \begin{array}{l} ** \text{ nonlinear, coupled} \\ \text{element (function} \\ \text{of } X_A, \xi, t) . \end{array}$$

The integrated robot manipulator model (2.35) can be decomposed into an input decentralized form (3.2) since the integrated input matrix  $B_A(X_A, \xi, t)$  is equivalent to the augmented actuator matrix which is readily in the desired block diagonal form. One drawback of this description is that the nonzero elements of the interconnection matrix  $A_{Aij}(X_A, \xi, t)$ , for  $i \neq j$ , are not in the image of the  $i$ th input matrix  $B_{Ai}$ .

### 3.2.2 Decomposition Based On Integrated Model With Joint Angle, Velocity And Acceleration As State Variables

In this subsection, the decomposition of the integrated robot manipulator dynamic model represented by equations (2.50), (2.51), and (2.52) is considered. In order to apply a decentralized and/or hierarchical control strategy, the robot manipulator dynamic model is decomposed into an input decentralized form.



Let the integrated system matrix  $A_B(X_B, \xi, t)$  of equation (2.51), and the integrated input matrix  $B_B(X_B, \xi, t)$  of equation (2.52) be partitioned in the following form :

$$A_B(X_B, \xi, t) = \left[ \begin{array}{c|c|c|c} A_{B11}(X_B, \xi, t) & A_{B12}(X_B, \xi, t) & \dots & A_{B1N}(X_B, \xi, t) \\ \hline - & - & - & - \\ A_{B21}(X_B, \xi, t) & A_{B22}(X_B, \xi, t) & \dots & A_{B2N}(X_B, \xi, t) \\ \hline - & - & - & - \\ \vdots & \vdots & \ddots & \vdots \\ \hline - & - & - & - \\ A_{BN1}(X_B, \xi, t) & A_{BN2}(X_B, \xi, t) & \dots & A_{BNN}(X_B, \xi, t) \\ \hline \end{array} \right], \quad (3.5)$$

and

$$B_B(X_B, \xi, t) = \left[ \begin{array}{c|c|c|c} B_{B11}(X_B, \xi, t) & B_{B12}(X_B, \xi, t) & \dots & B_{B1N}(X_B, \xi, t) \\ \hline - & - & - & - \\ B_{B21}(X_B, \xi, t) & B_{B22}(X_B, \xi, t) & \dots & B_{B2N}(X_B, \xi, t) \\ \hline - & - & - & - \\ \vdots & \vdots & \ddots & \vdots \\ \hline - & - & - & - \\ B_{BN1}(X_B, \xi, t) & B_{BN2}(X_B, \xi, t) & \dots & B_{BNN}(X_B, \xi, t) \\ \hline \end{array} \right], \quad (3.6)$$

where

$A_{Bij}(X_B, \xi, t)$  : 3x3 matrix

$B_{Bij}(X_B, \xi, t)$  : 3x1 matrix

$i, j = 1, 2, \dots, N.$

With the above partitioning, the robot manipulator dynamic model can be written in the input decentralized form which is as follows :

$$\dot{X}_{Bi}(t) = A_{Bii}(X_B, \xi, t) X_{Bi}(t) + B_{Bii}(X_B, \xi, t) U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{Bij}(X_B, \xi, t) X_j(t) + \sum_{\substack{j=1 \\ j \neq i}}^N B_{Bij}(X_B, \xi, t) U_j(t) , \quad (3.7)$$

$$X_{Bi}(t) = \begin{bmatrix} \theta_i & \dot{\theta}_i & \ddot{\theta}_i \end{bmatrix}^T , \quad (3.8)$$

$$X_{Bi}(t) \in \mathbb{R}^3, \quad i \in \mathcal{J} ,$$

where

- $X_{Bi}(t)$  :  $i$ th component of the  $X_B(t)$  vector  $\equiv$  state vector of the  $i$ th actuator
- $X_{Bj}(t)$  :  $j$ th component of the  $X_B(t)$  state vector
- $A_{Bii}(X_B, \xi, t)$  : 3x3  $i$ th diagonal submatrix of  $A_B(X_B, \xi, t)$
- $A_{Bij}(X_B, \xi, t)$  : 3x3  $ij$ th off-diagonal submatrix of  $A_B(X_B, \xi, t)$
- $B_{Bii}(X_B, \xi, t)$  : 3x1  $i$ th diagonal submatrix of  $B_B(X_B, \xi, t)$
- $B_{Bij}(X_B, \xi, t)$  : 3x1  $ij$ th off-diagonal submatrix of  $B_B(X_B, \xi, t)$
- $U_i(t)$  :  $i$ th row of input vector  $U(t) \equiv$  input to the  $i$ th actuator
- $U_j(t)$  :  $j$ th row of input vector  $U(t) \equiv$  input to the  $j$ th actuator.

For non-direct drive robot manipulators, the magnitudes of the non-zero elements of the off-block diagonal submatrices in  $B_B(X_B, \xi, t)$  are often very small compared to the elements of the diagonal submatrices  $B_{Bii}(X_B, \xi, t)$ . Thus, the off-block diagonal submatrices  $B_{Bij}(X_B, \xi, t)$  can often be assumed to be negligible and can be ignored and, hence, in these situations, the integrated input matrix  $B_B(X_B, \xi, t)$  may be treated as in block diagonal form, that is,

$$B_B(X_B, \xi, t) = \text{diag} [ B_{B11}(X_B, \xi, t) , B_{B22}(X_B, \xi, t) , \dots , B_{BNN}(X_B, \xi, t) ]. \quad (3.9)$$

Then, in this case, the interconnected robot manipulator model can be written as

$$\dot{X}_{Bi}(t) = A_{Bii}(X_B, \xi, t) X_{Bi}(t) + B_{Bii}(X_B, \xi, t) U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{Bij}(X_B, \xi, t) X_j(t) , \quad (3.10)$$

without the last term on the right hand side of equation (3.7). As an example, for the robotic system considered in Chapter 2, the non-zero element of the off-diagonal submatrices has the value of  $-0.00639 \pm 0.007839$ , which can be considered negligible compared to the value of  $0.6248 \pm 0.01558$  for the smallest non-zero element of the diagonal submatrices, hence the off-diagonal submatrices in the integrated input matrix  $B_B(X_B, \xi, t)$  in equation (3.6) can be ignored.

However, if the above assumption is not valid, such as, in the case of direct drive robots in general, then one has to use expression (3.7) to represent the interconnected robot manipulator system. In this study, only non-direct drive robot manipulator is considered, and it is assumed that equation (3.9) hold.

In general, equation (3.10) describes the dynamic model of the  $i$ th subsystem which is interconnected with the other subsystems through  $A_{Bij}(X_B, \xi, t)X_{Bj}(t)$ . The structure of the  $A_{Bii}(X_B, \xi, t)$ ,  $A_{Bij}(X_B, \xi, t)$ , and  $B_{Bii}(X_B, \xi, t)$  matrices may be obtained as follow :

$$A_{Bii}(X_B, \xi, t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ ** & ** & ** \end{bmatrix} , \quad B_{Bii}(X_B, \xi, t) = \begin{bmatrix} 0 \\ 0 \\ ** \end{bmatrix} , \quad (3.11)$$

$$A_{Bij}(X_B, \xi, t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ ** & ** & ** \end{bmatrix},$$

\*\* nonlinear, coupled element (function of  $X_B, \xi, t$ ) .

The main advantage of this representation of the robot manipulator dynamic model is that all the nonlinear, coupled and time varying elements of the system are in the image of the input submatrices, as shown by equation (3.11). This is important in the sense that the control input  $U_i(t)$  which enters the  $i$ th subsystem through the input submatrix  $B_{Bii}(X_B, \xi, t)$  can affect and compensate the nonlinear, coupled and time varying elements directly.

### 3.3 ROBOT MANIPULATOR AS AN UNCERTAIN INTERCONNECTED SYSTEM

In this section, it will be shown that the robot manipulator dynamic model in the above descriptions can be transformed into a linear interconnected system with uncertainties representation. This representation of the robot manipulator is required in order to design the tracking controller based on a deterministic approach which will be presented in the next few chapters.

In general, the robot manipulator model in input decentralized form as described by equation (3.2), or in the form of equation (3.10), can be represented as

$$\dot{X}_i(t) = A_i(X, \xi, t) X_i(t) + B_i(X, \xi, t) U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t) . \quad (3.12)$$

Since the physical parameters of the actuators and the manipulator (length and mass of its links), the range of its payload, the joint displacements as well as the range of the velocities are known (specified by the manufacturer), the bounds on the elements of the matrices  $A_i(X, \xi, t)$ ,  $B_i(X, \xi, t)$ , and  $A_{ij}(X, \xi, t)$  can be computed and specified in the form :

$$\begin{aligned}\underline{a}_{ij}^i &\leq a_{ij}^i(X, \xi, t) \leq \bar{a}_{ij}^i \\ \underline{b}_i^i &\leq b_i^i(X, \xi, t) \leq \bar{b}_i^i \\ \underline{a}_{ij}^{ij} &\leq a_{ij}^{ij}(X, \xi, t) \leq \bar{a}_{ij}^{ij},\end{aligned}\tag{3.13}$$

where

$a_{ij}^i(X, \xi, t)$  :  $ij$ th element of the  $A_i(X, \xi, t)$  matrix

$b_i^i(X, \xi, t)$  :  $i$ th row of the  $B_i(X, \xi, t)$  matrix

$a_{ij}^{ij}(X, \xi, t)$  :  $ij$ th element of the  $A_{ij}(X, \xi, t)$  matrix ,

and the upper and lower bars indicate the maximum and minimum values, respectively. Since these bounds are known, the matrices  $A_i(X, \xi, t)$ ,  $B_i(X, \xi, t)$  can be written as :

$$\begin{aligned}A_i(X, \xi, t) &= A_i + \Delta A_i(X, \xi, t), \\ B_i(X, \xi, t) &= B_i + \Delta B_i(X, \xi, t),\end{aligned}\tag{3.14}$$

where  $A_i$  and  $B_i$ , which are time-invariant, represent the nominal part of  $A_i(X, \xi, t)$  and  $B_i(X, \xi, t)$  respectively. The elements of the matrices  $A_i$  and  $B_i$ , respectively, may be determined as follows :

$$\begin{aligned}
a_{ij}^i &= \frac{\underline{a}_{ij}^i + \overline{a}_{ij}^i}{2} \\
b_i^i &= \frac{\underline{b}_i^i + \overline{b}_i^i}{2} .
\end{aligned}
\tag{3.15}$$

The elements of the matrices  $\Delta A_i(X, \xi, t)$  and  $\Delta B_i(X, \xi, t)$ , respectively, can be obtained as

$$\begin{aligned}
\Delta a_{ij}^i(X, \xi, t) &= a_{ij}^i(X, \xi, t) - a_{ij}^i \\
\Delta b_i^i(X, \xi, t) &= b_i^i(X, \xi, t) - b_i^i ,
\end{aligned}
\tag{3.16}$$

where

$\Delta a_{ij}^i(X, \xi, t)$ :  $ij$ th element of the  $\Delta A_i(X, \xi, t)$  matrix

$\Delta b_i^i(X, \xi, t)$ :  $i$ th row of the  $\Delta B_i(X, \xi, t)$  matrix ,

and their range of variations can be computed as follows :

$$\begin{aligned}
-r_{ij}^i &\leq \Delta a_{ij}^i(X, \xi, t) \leq r_{ij}^i , \\
-s_i^i &\leq \Delta b_i^i(X, \xi, t) \leq s_i^i ,
\end{aligned}
\tag{3.17}$$

where

$$r_{ij}^i = \overline{a}_{ij}^i - a_{ij}^i , \quad s_i^i = \overline{b}_i^i - b_i^i .$$

These values are computed off-line, and may be calculated only once for the robotic system. That is, equations (3.15) and (3.17) need not be recalculated for different manipulator payload, task or trajectory if all the possible range of the payload, configurations (work space) of the manipulator as well as its maximum velocity, etc. have been taken into account in determining the bounds on the elements of the matrices  $A_i(X, \xi, t)$ ,  $B_i(X, \xi, t)$ , and  $A_{ij}(X, \xi, t)$  in equation (3.13). However, if the bounds in equation (3.13) are obtained based only on a specific trajectory or task, then equations (3.15) and (3.17) have to be recalculated if the manipulator is needed to perform a new task or follow a new trajectory. In determining these values, a full dynamic model of the robot manipulator in the

form of equation (3.12) is required.

In view of equations (3.13) - (3.17), without loss of generality, equation (3.12) can be rewritten as

$$\dot{X}_i(t) = [A_i + \Delta A_i(X, \xi, t)] X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t), \quad (3.18)$$

where the elements of the matrices  $\Delta A_i(X, \xi, t)$ ,  $\Delta B_i(X, \xi, t)$ , and  $A_{ij}(X, \xi, t) - \Delta a_{ij}^i(X, \xi, t)$ ,  $\Delta b_i^i(X, \xi, t)$ , and  $a_{ij}^{ij}(X, \xi, t)$ , respectively – are considered as uncertainties which belong to uncertainty bounding sets  $\mathcal{R}$ ,  $\mathcal{Y}$ , and  $\mathcal{V}$ , respectively. The uncertainty bounding sets are defined as follows :

$$\begin{aligned} \mathcal{R} &\triangleq \{ \Delta a_{ij}^i ; \forall i \in \mathcal{I}, \forall j \in \mathcal{J} \mid -r_{ij}^i \leq \Delta a_{ij}^i \leq r_{ij}^i \} \\ \mathcal{Y} &\triangleq \{ \Delta b_i^i ; \forall i \in \mathcal{I}, \forall j \in \mathcal{J} \mid -s_i^i \leq \Delta b_i^i \leq s_i^i \} \\ \mathcal{V} &\triangleq \{ a_{ij}^{ij} ; \forall i \in \mathcal{I}, \forall j \in \mathcal{J} \mid \underline{a}_{ij}^{ij} \leq a_{ij}^{ij} \leq \bar{a}_{ij}^{ij} \} , \end{aligned} \quad (3.19)$$

where the values of  $r_{ij}^i$ ,  $s_i^i$ ,  $\underline{a}_{ij}^{ij}$ , and  $\bar{a}_{ij}^{ij}$  are as given in (3.13) and (3.17). Thus the nonlinear coupled integrated robot manipulator dynamic model (equation 3.12) is reduced to a set of interconnected linear subsystems with bounded uncertainties described by equation (3.18) and (3.19).

It should be noted that equation (3.18) is a continuous function of time  $t$ , and highly nonlinear and coupled.

For a robot manipulator in the form of equation (3.2), the matrices  $A_i$ ,  $B_i$ ,  $\Delta A_i(*)$ ,  $\Delta B_i(*)$ , and  $A_{ij}(*)$  can be shown to have the following form :

$$A_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & a_{22}^i & a_{23}^i \\ 0 & a_{32}^i & a_{33}^i \end{bmatrix}, \quad B_i = \begin{bmatrix} 0 \\ 0 \\ b_3^i \end{bmatrix},$$

$$\Delta A_i(*) = \begin{bmatrix} 0 & 0 & 0 \\ \Delta a_{21}^i(*) & \Delta a_{22}^i(*) & \Delta a_{23}^i(*) \\ 0 & 0 & 0 \end{bmatrix}, \quad (3.20)$$

$$A_{ij}(*) = \begin{bmatrix} 0 & 0 & 0 \\ a_{21}^{ij}(*) & a_{22}^{ij}(*) & a_{23}^{ij}(*) \\ 0 & 0 & 0 \end{bmatrix}, \quad \Delta B_i(*) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

where  $(*)$  represents the arguments  $(X_A, \xi, t)$ . It can be seen from the structure of the  $A_{ij}(*)$  and  $\Delta A_i(*)$  matrices in equation (3.20) that the uncertainties lies outside the range space of the input matrix  $B_i$ .

For a robot manipulator integrated dynamic model in the form of equation (3.10), it can be shown that the matrices  $A_i$ ,  $B_i$ ,  $\Delta A_i(*)$ ,  $\Delta B_i(*)$ , and  $A_{ij}(*)$  have the following form :

$$A_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a_{31}^i & a_{32}^i & a_{33}^i \end{bmatrix}, \quad B_i = \begin{bmatrix} 0 \\ 0 \\ b_3^i \end{bmatrix},$$



$$\Delta A_i(*) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \Delta a_{31}^i(*) & \Delta a_{32}^i(*) & \Delta a_{33}^i(*) \end{bmatrix}, \quad (3.21)$$

$$A_{ij}(*) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ a_{31}^{ij}(*) & a_{32}^{ij}(*) & a_{33}^{ij}(*) \end{bmatrix}, \quad \Delta B_i(*) = \begin{bmatrix} 0 \\ 0 \\ \Delta b_3^i(*) \end{bmatrix},$$

where  $(*)$  represents the arguments  $(X_A, \xi, t)$ .

For the  $i$ th subsystem, equation (3.21) shows that the uncertainties presents in the system and input matrix,  $\Delta A_i(X_A, \xi, t)$  and  $\Delta B_i(X_A, \xi, t)$  respectively, and the interconnection matrix  $A_{ij}(X_A, \xi, t)$  lie in the range space (image) of the nominal input matrix  $B_i$ . Thus, the control input which enters through the input matrix can compensates the uncertainties present in  $A_i(X_A, \xi, t)$  and  $B_i(X_A, \xi, t)$ , and the interaction functions due to  $A_{ij}(X_A, \xi, t)$  matrix.

It should be noted that, if the uncertainties and the interaction function lies outside the range space of the input matrix, there is no control input that can compensate for them even if the uncertainties and the nonlinear interconnection functions are known [Chen, 1987b; Fu and Liao, 1990]. Due to this reason, the robot manipulator model in the form of equation (3.10) should be used in designing a robust controller for the system. In other words, in deriving the mathematical model of the robot manipulator for control purposes, the joint angles, velocities, and the acceleration should be chosen as the state variables, instead of the joint angles, velocities, and the armature currents.

In the following section, the integrated model of the three dof robot manipulator as derived in Chapter 2 is decomposed and transformed into the uncertain input decentralized description as presented above.

### 3.4 APPLICATION TO A THREE dof ROBOT MANIPULATOR

In this section, based on the procedures described in Section 3.2.2 and Section 3.3, the three dof revolute robot manipulator integrated dynamic model is transformed and reduced into a set of interconnected subsystems with bounded uncertainties description.

Since the robot manipulator considered is of three dof, the robot manipulator's dynamic model as presented by equation (2.75) can be decomposed into three interconnected subsystems. Based on equations (3.5) and (3.6), the integrated system matrix  $A(X_B, \xi, t)$ , and the input matrix  $B(X_B, \xi, t)$ , equations (2.76) and (2.77), respectively, can be partitioned as follow (for convenience the subscript  $B$  has been dropped) :

$$A(X, \xi, t) = \left[ \begin{array}{ccc|ccc|ccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{32} & a_{33} & 0 & a_{35} & a_{36} & 0 & a_{38} & a_{39} \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} & a_{67} & a_{68} & a_{69} \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_{92} & a_{93} & a_{94} & a_{95} & a_{96} & 0 & a_{98} & a_{99} \end{array} \right], \quad (3.22)$$

$$B(X, \xi, t) = \begin{bmatrix} 0 & | & 0 & | & 0 \\ 0 & | & 0 & | & 0 \\ b_{11} & | & 0 & | & 0 \\ \hline 0 & | & 0 & | & 0 \\ 0 & | & 0 & | & 0 \\ 0 & | & b_{22} & | & b_{23} \\ \hline 0 & | & 0 & | & 0 \\ 0 & | & 0 & | & 0 \\ 0 & | & b_{32} & | & b_{33} \end{bmatrix} . \quad (3.23)$$

Then the integrated three dof robot manipulator model (equation 2.75) can be rewritten in the following input decentralized form :

$$\dot{X}_i(t) = A_i(X, \xi, t) X_i(t) + B_i(X, \xi, t) U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^3 A_{ij}(X, \xi, t) X_j(t) + \sum_{\substack{j=1 \\ j \neq i}}^3 B_{ij}(X, \xi, t) U_j(t) , \quad (3.24)$$

$$X_i(t) = \begin{bmatrix} \theta_i & , & \dot{\theta}_i & , & \ddot{\theta}_i \end{bmatrix}^T , \quad (3.25)$$

$$i = j = 1, 2, 3 ,$$

where

$X_i(t)$  :  $i$ th component of the  $X(t)$  vector  $\equiv$  state vector of the  $i$ th actuator

$X_j(t)$  :  $j$ th component of the  $X(t)$  state vector  $\equiv$  state vector of the  $j$ th actuator

- $A_i(X, \xi, t)$  : 3x3  $i$ th diagonal submatrix of  $A(X, \xi, t)$   
 $A_{ij}(X, \xi, t)$  : 3x3  $ij$ th off-diagonal submatrix of  $A(X, \xi, t)$   
 $B_i(X, \xi, t)$  : 3x1  $i$ th diagonal submatrix of  $B(X, \xi, t)$   
 $B_{ij}(X, \xi, t)$  : 3x1  $ij$ th off-diagonal submatrix of  $B(X, \xi, t)$   
 $U_i(t)$  :  $i$ th row of input vector  $U(t) \equiv$  input to the  $i$ th actuator  
 $U_j(t)$  :  $j$ th row of input vector  $U(t) \equiv$  input to the  $j$ th actuator,

and where, the structure of the  $A_i(X, \xi, t)$ ,  $B_i(X, \xi, t)$ ,  $A_{ij}(X, \xi, t)$  and  $B_{ij}(X, \xi, t)$  matrices are in the form of equation (3.11).

Based on the physical parameters of the actuators and the manipulator as given in Table 2.1 and Table 2.2, as well as the allowable range of operation and the maximum allowable load as specified in Chapter 2, the bounds on the nonzero elements of the matrices  $A_i(X, \xi, t)$ ,  $B_i(X, \xi, t)$ ,  $A_{ij}(X, \xi, t)$ , and  $B_{ij}(X, \xi, t)$  can be computed. Then, based on equations (3.15) and (3.17), the nominal matrices  $A_i$ ,  $B_i$ , as well as the bounds on the non-zero elements of the matrices  $\Delta A_i(X, \xi, t)$ ,  $\Delta B_i(X, \xi, t)$ ,  $A_{ij}(X, \xi, t)$  and  $B_{ij}(X, \xi, t)$  in (3.24) for each subsystem can be calculated off-line as follows :

$$\begin{aligned}
 A_1 &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -87.8068 & -10.9546 \end{bmatrix}, & B_1 &= \begin{bmatrix} 0 \\ 0 \\ 0.6987 \end{bmatrix} \\
 A_2 &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -88.1354 & -10.9543 \end{bmatrix}, & B_2 &= \begin{bmatrix} 0 \\ 0 \\ 0.6248 \end{bmatrix} \\
 A_3 &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -89.759 & -10.9768 \end{bmatrix}, & B_3 &= \begin{bmatrix} 0 \\ 0 \\ 0.6349 \end{bmatrix}
 \end{aligned}$$

$$r_{32}^1(*) = 2.8768 \quad ; \quad r_{33}^1(*) = 0.3683 \quad (3.26)$$

$$r_{31}^2(*) = 17.5617 \quad ; \quad r_{32}^2(*) = 2.46 \quad ; \quad r_{33}^2(*) = 0.09303$$

$$r_{31}^3(*) = 7.9962 \quad ; \quad r_{32}^3(*) = 0.2296 \quad ; \quad r_{33}^3(*) = 0.001653$$

$$s_3^1(*) = 0.02289 \quad ; \quad s_3^2(*) = 0.01558 \quad ; \quad s_3^3(*) = 0.000166$$

$$-1.5936 \leq a_{32}^{12}(*) \leq 1.5422 \quad ; \quad -0.1372 \leq a_{33}^{12}(*) \leq 0.1372$$

$$-0.8856 \leq a_{32}^{13}(*) \leq 0.91 \quad ; \quad -0.07876 \leq a_{33}^{13}(*) \leq 0.08055$$

$$-0.601 \leq a_{32}^{21}(*) \leq 0.6022 \quad ; \quad -0.1054 \leq a_{33}^{21}(*) \leq 0.1054$$

$$-0.176 \leq a_{31}^{23}(*) \leq 0.176 \quad ; \quad -0.6671 \leq a_{32}^{23}(*) \leq 2.4571$$

$$-0.0855 \leq a_{33}^{23}(*) \leq 0.1309 \quad ; \quad -0.3676 \leq a_{32}^{31}(*) \leq 0.3444$$

$$-0.06452 \leq a_{33}^{31}(*) \leq 0.06234 \quad ; \quad -0.4037 \leq a_{31}^{32}(*) \leq 0.4037$$

$$-0.4379 \leq a_{32}^{32}(*) \leq 2.2381 \quad ; \quad -0.04044 \leq a_{33}^{32}(*) \leq 0.05323$$

$$-0.01423 \leq b_{32}^{23}(*) = b_{32}^{32}(*) \leq 0.001449$$

The non-zero elements of the off-block diagonal input submatrices  $B_{23}(X, \xi, t)$  and  $B_{32}(X, \xi, t)$  have the value of  $-0.00639 \pm 0.007839$ , which is assumed to be negligible compared to the non-zero element of the diagonal submatrices. Thus, they can be ignored. Thus, the robot manipulator model can be decomposed into three interconnecting subsystems with bounded uncertainties in the form of equation (3.18). That is, equation (3.24) can be rewritten as

$$\begin{aligned} \dot{X}_i(t) = & [A_i + \Delta A_i(X, \xi, t)] X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) + \\ & \sum_{\substack{j=1 \\ j \neq i}}^3 A_{ij}(X, \xi, t) X_j(t) \end{aligned} \quad (3.27)$$

$$i = j = 1, 2, 3.$$

It is clear from the above equations that the structure of the components of equation (3.27) are in the form of equation (3.21). Thus, a robust decentralized and/or hierarchical tracking controller which can overcome the uncertainties, nonlinearities, and couplings may be designed for the robot manipulator based on a deterministic approach.

The three dof robot manipulator model represented by equation (3.27) with the values as given in equation (3.26) will be used in deriving the decentralized and/or hierarchical tracking controller in this study. It should be emphasized here that the above description is only used to derive the decentralized and/or hierarchical controller for the robot manipulator, while the integrated robot manipulator model (equation 2.75) is used to represent the real three dof robot manipulator in the simulations in the following chapters.

### 3.5 CONCLUSION

In this chapter, the robot manipulator is treated as a large scale uncertain system. A procedure to decompose and transform a robot manipulator system into a set of interconnected subsystems with bounded uncertainties is presented. The problem of selecting the appropriate state space representation of the interconnected robot manipulator subsystems is critical for designing a robust decentralized and hierarchical tracking controller for the system, based on a deterministic approach. It was shown in this chapter that by choosing an appropriate set of state variables for the robot manipulator dynamic model, the uncertainties present in the subsystems system and input matrices, as well as the

interconnection functions can be placed in the range space of the subsystems input matrix. This will enable the control input which enters the subsystem through the input matrix to affect and compensates for the uncertainties and the nonlinear interaction functions. The decomposition of the robot manipulator into an input decentralized form with bounded uncertainties description presented in this chapter will be the basis for the decentralized and hierarchical tracking controller formulations, based on a deterministic approach, which will be presented in the following chapters.

## CHAPTER 4

# DECENTRALIZED TRACKING CONTROLLER DESIGN FOR ROBOT MANIPULATORS

### 4.1 INTRODUCTION

Decentralized control strategies are increasingly being used in the design of robotic controllers due to economical and practical reasons. The majority of the current industrial robot manipulators utilize a decentralized control method, where each joint is treated individually as a simple linear servomechanism with, for example, proportional plus integral plus derivative (PID) controllers. In designing these controllers, the highly nonlinear, coupled and time-varying dynamics of the robot manipulator linkage have usually been completely ignored. Thus, the method is satisfactory only for an industrial robot manipulator designed with less demanding path control application, since the influence of the nonlinear, coupled and time-varying dynamics is strong for fast manipulator motion.

The application of a decentralized control concept at the executive level for a robotic system has been considered by several researchers [Vukobratovic and Stokic, 1983; Stokic and Vukobratovic, 1984; and Pandian et.al., 1988]. In these papers, the robotic system is treated as a set of subsystems, each represented by an actuator model (equation 2.21), interconnected through the dynamics of the mechanical part of the robotic system (equation 1.1). For each subsystem, a completely decentralized controller is designed based only on the local states of the subsystem. The controller consists of two parts : programmed nominal controller and compensating controller. Based on the decoupled free subsystem model, the programmed nominal controller is designed to track the nominal (desired) trajectories in the absence of the couplings. A compensating controller is



then designed based on the deviation model of the subsystem from the nominal states (trajectories). Pandian et.al. [1988] proposed a high gain control technique to design the compensating controller such that the deviation model tends to zero asymptotically. In Vukobratovic and Stokic [1983], and Stokic and Vukobratovic [1984], the compensating controller is obtained using an optimal control technique. It was shown that the decentralized controller will render the overall robot manipulator practically stable if a given sufficient condition is satisfied.

In this study, robust tracking controllers which are capable of withstanding all the expected variations and uncertainties in the system are presented based on a class of deterministic approach developed by Leitmann [1981], and Corless and Leitmann [1981]. A brief description of the approach is presented next.

## 4.2 DETERMINISTIC CONTROL OF UNCERTAIN SYSTEMS

Consider an uncertain system where all the uncertainties can be 'lumped' together as described by the following equation :

$$\dot{X}(t) = AX(t) + BU(t) + Be(X, t) \quad ; \quad X \in \mathbb{R}^n \quad (4.1)$$

where  $e(X, t)$  is the 'lumped' uncertain elements.

The control of such a system can in general be treated either by using stochastic theory, or the use of deterministic techniques where the uncertainties are described only in terms of their bounds. No statistical information is assumed about the uncertainties.

Among the deterministic approaches that can be used to control such a system are the techniques based on :

1. Variable Structure method [Young, 1978]
2. The second method of Lyapunov in which a Lyapunov function is minimized with respect to the control variables. This will leads to a class of 'min-max' controllers [Leitmann, 1979; Gutman and Palmor,

1982]

3. The class of controllers proposed by Leitmann [1981], and Corless and Leitmann [1981].

The first two classes of controllers guarantee asymptotic stability of the system behaviour. However, the controllers are discontinuous in structure, thus, they are difficult to be realized in practice.

The class of controllers proposed by Leitmann [1981], and Corless and Leitmann [1981] also uses the second method of Lyapunov. It is based on the theory of guaranteed stability (or ultimate boundedness) of the solution of the uncertain system (4.1). The control approximates the 'min-max' control by a single-valued continuous function in the neighbourhood of the switching surface. Thus, the control action is continuous everywhere. However, the behaviour of the system obtained is generally ultimate boundedness, and not asymptotic stability.

In the following, the deterministic control approach of Leitmann [1981], and Corless and Leitmann [1981] is briefly described.

#### 4.2.1 Deterministic Control Approach of Leitmann [1981], and Corless and Leitmann [1981]

In the ultimate boundedness results of the mentioned literatures for an uncertain system represented by equation (4.1), a nonlinear saturated feedback control law is used. The feedback controller is based solely on the knowledge of the maximum possible value of the norm of the 'lumped' uncertainties  $e(X, t)$ . This value may depend on  $X$  and  $t$ . The controller proposed is of the form  $U(t)=\Phi(X,t)$ , where for a given  $\epsilon > 0$ ,

$$\Phi(X, t) = \begin{cases} -\frac{B^T P X}{\|B^T P X\|} \rho(X, t) & \text{if } \|B^T P X\| > \epsilon \\ \|\Phi(Z, t)\| \leq \rho(X, t) & \text{if } \|B^T P X\| \leq \epsilon \end{cases} \quad (4.2)$$

where  $P$  is the positive definite solution of the Lyapunov equation

$$P A + A^T P = - Q \quad (4.3)$$

for a positive definite symmetric matrix  $Q$ , and where the system matrix  $A$  is assumed to be a stable matrix.  $\rho(X, t)$  is the norm bound of the 'lumped' uncertainties :

$$\|e(X, t)\| \leq \rho(X, t) . \quad (4.4)$$

Beginning with the Lyapunov function

$$\mathcal{L} = X^T P X , \quad (4.5)$$

the derivative of  $\mathcal{L}$  along the solution of the system (4.1) and (4.2) can be shown to be bounded as [Leitmann, 1981] :

$$\dot{\mathcal{L}} \leq -\alpha_2 \|X\|^2 + \alpha_1 \|X\| + \alpha_0 \quad (4.6)$$

where  $\alpha_i \geq 0$  ( $i=0,1,2$ ) are some coefficients of  $\|X\|^i$ .

Since  $\alpha_2 > 0$ , consequently,  $\dot{\mathcal{L}} < 0$  for all  $t$  and  $X \in \mathfrak{B}^c(\eta)$ , where  $\mathfrak{B}^c(\eta)$  is the complement of the closed ball  $\mathfrak{B}(\eta)$ , centred at  $X=0$  with radius

$$\eta = \frac{\alpha_1 + \sqrt{\alpha_1^2 + 4\alpha_2\alpha_0}}{2\alpha_2} . \quad (4.7)$$

Since  $P$  is positive definite, the following ellipsoids can be defined :

$$\mathfrak{E}(\kappa) \triangleq \left\{ X \in \mathbb{R}^n \mid X^T P X \leq \kappa = \text{constant} > 0 \right\} . \quad (4.8)$$

Let the smallest Lyapunov ellipsoid that contains the closed ball  $\mathfrak{B}(\eta)$  be  $\mathfrak{E}(\underline{\kappa})$ , where in view of

$$\lambda_{\min}(P) \|X\|^2 \leq X^T P X \leq \lambda_{\max}(P) \|X\|^2, \quad (4.9)$$

and  $\lambda_{\min}(P) > 0$  since  $P > 0$ ,  $\underline{\kappa}$  is given by

$$\underline{\kappa} = \lambda_{\max}(P) \eta^2. \quad (4.10)$$

Then, given a  $Q > 0$ , it was shown in Leitmann [1981] that the uncertain system (4.1), with the nonlinear saturating controller  $\Phi(X, t)$  and corresponding to an initial condition  $(X_0, t_0)$ , has uniform ultimate boundedness with respect to the set  $\mathfrak{X}(\underline{\kappa})$ , where

$$\mathfrak{X}(\underline{\kappa}) \triangleq \{ X \in \mathbb{R}^n \mid X^T P X < \underline{\kappa} \}. \quad (4.11)$$

Uniform ultimate boundedness means that the solution of the system (4.1) with the saturating controller will enter an ultimate region after a finite time and remains there thereafter. Hence, the closed loop system is practically stable.

#### 4.2.2 Application Of A Deterministic Approach To Robot Manipulator Control – An Overview

The implementation of the deterministic approach for robot manipulator tracking control problems has been considered by Corless et.al. [1984], Chen [1987b], Chen and Eyo [1988], Corless [1989], Chen and Pandey [1990], and Shoureshi et.al. [1987; 1990], for example. In most of these papers, the actuator dynamics which constitute an integral part of a robot manipulator have been ignored; the controller is designed based only on the dynamics of the mechanical part of the robot manipulator. Furthermore, in some of the approaches [Corless et.al., 1984; Chen and Eyo, 1988; Shoureshi et.al., 1987; 1990], the controller requires a nonlinear compensator which is based on the on-line computation of the complicated robot manipulator dynamics. Chen and Pandey [1990] used a Computed Torque Technique coupled with the deterministic approach to design a robust hybrid controller for a robot manipulator. Similar to the above mentioned methods, the controller requires on-line computation of the robot manipulator

dynamic equations. In all of these works, the proposed controllers are centralized in structure. Thus, the methods are usually complex and demand heavy on-line computation of the robot manipulator dynamics.

In this chapter, a decentralized tracking controller for robot manipulators is presented. The decentralized controller is formulated based on the results of [Leitmann, 1981; Corless and Leitmann, 1981; Chen, 1986; 1987a; 1988], but without the above mentioned computation difficulties. Each local controller is designed based only on the local states of the corresponding subsystem. A complete model of the robot manipulator dynamics is used in designing the controllers. It is assumed that the bounds on the nonlinearities, couplings and uncertainties present in the system are known. The decentralized tracking controllers are nonlinear and are based only on these bounds. It will be shown that, the controller will render the errors between the robot manipulator responses and the corresponding reference trajectories practically stable, and track the desired trajectories, in spite of the uncertainties, couplings and nonlinearities present in the system.

The formulation of the decentralized tracking problem and some standard assumptions are presented in the following section.

### 4.3 PROBLEM FORMULATION

As shown in Chapter 3, robot manipulator dynamics can be represented by a set of interconnected linear subsystems with bounded uncertainties described by equations (3.18) and (3.19) which are as the following :

$$\begin{aligned} \dot{X}_i(t) = & [A_i + \Delta A_i(X, \xi, t)] X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) + \\ & \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t), \end{aligned} \quad (4.12)$$

$$X_i(t) = [ \theta_i(t), \dot{\theta}_i(t), \ddot{\theta}_i(t) ]^T, \quad (4.13)$$

$$X_i(t_o) = X_{i_o} ; \quad i \in \mathcal{I}, \quad (4.14)$$

where  $X_{i_o}$  is a given initial condition vector. The elements of the  $\Delta A_i(X, \xi, t)$ ,  $\Delta B_i(X, \xi, t)$ , and  $A_{ij}(X, \xi, t)$  matrices,  $\Delta a_{ij}^i(X, \xi, t)$ ,  $\Delta b_i^i(X, \xi, t)$ , and  $a_{ij}^{ij}(X, \xi, t)$ , respectively, may be considered as uncertainties which belong to uncertainty bounding sets  $\mathcal{R}$ ,  $\mathcal{Y}$ , and  $\mathcal{V}$  respectively. The uncertainty bounding sets may be defined as follows :

$$\begin{aligned} \mathcal{R} &\triangleq \{ \Delta a_{ij}^i(*) ; \forall i \in \mathcal{I}, \forall j \in \mathcal{I} \mid -r_{ij}^i \leq \Delta a_{ij}^i(*) \leq r_{ij}^i \} \\ \mathcal{Y} &\triangleq \{ \Delta b_i^i(*) ; \forall i \in \mathcal{I}, \forall j \in \mathcal{I} \mid -s_i^i \leq \Delta b_i^i(*) \leq s_i^i \} \\ \mathcal{V} &\triangleq \{ a_{ij}^{ij}(*) ; \forall i \in \mathcal{I}, \forall j \in \mathcal{I} \mid \underline{a}_{ij}^{ij} \leq a_{ij}^{ij}(*) \leq \bar{a}_{ij}^{ij} \} \end{aligned} \quad (4.15)$$

where the values of  $r_{ij}^i$ ,  $s_i^i$ ,  $\underline{a}_{ij}^{ij}$ , and  $\bar{a}_{ij}^{ij}$  are as given in (3.13) and (3.17). For robot manipulators, these bounds are known. Equation (4.12), which represents the complete dynamics of the robotic arm is a continuous function of the states and time  $t$ . So are the uncertain elements.

To reduce the effect of the uncertainties on the performance of the robot manipulator, an appropriate control law which takes into account the uncertainties must be formulated.

Let a continuous function  $X_d(t) \in \mathcal{R}^{3N}$  be the desired state trajectory, where  $X_d(t)$  is defined as :

$$\begin{aligned} X_d(t) &= [ X_{d1}^T(t), X_{d2}^T(t), \dots, X_{dN}^T(t) ]^T \\ X_{di}(t) &= [ \theta_{di}(t), \dot{\theta}_{di}(t), \ddot{\theta}_{di}(t) ]^T. \end{aligned} \quad (4.16)$$

$$X_{di}(t) \in \mathcal{R}^3 ; \quad i \in \mathcal{I}.$$

The robot manipulator control problem is then to design a decentralized controller for each subsystem such that the robot actual state trajectory  $X_i(t)$  tracks the nominal or reference state trajectory  $X_{di}(t)$  as closely as possible for all  $t$  in spite of the uncertainties and nonlinearities present in the system.

It is assumed that :

- i. The pair  $(A_i, B_i)$  is stabilizable,
- ii. There exist continuous functions  $H_i(X, \xi, t) \in \mathbb{R}^{1 \times 3}$ , and  $E_i(X, \xi, t) \in \mathbb{R}$ , such that for all  $X \in \mathbb{R}^N$  and all  $t$  :

$$\Delta A_i(X, \xi, t) = B_i H_i(X, \xi, t) \quad (4.17)$$

$$\Delta B_i(X, \xi, t) = B_i E_i(X, \xi, t) \quad (4.18)$$

$$\|E_i(X, \xi, t)\| < 1. \quad (4.19)$$

- iii. There exist a Lebesgue function  $\Omega_i(t) \in \mathbb{R}$ , which is integrable on bounded intervals, such that

$$\dot{X}_{di}(t) = A_i X_{di}(t) + B_i \Omega_i(t). \quad (4.20)$$

- iv. There exist an upper bound,  $\bar{g}_{ij}$ , of the norm of the interconnection matrix for each  $i$  and  $j$  ( $i \neq j$ ) :

$$\|A_{ij}(X, \xi, t)\| \leq \bar{g}_{ij}. \quad (4.21)$$

In studying a decentralized large scale system, the question of whether the system can be stabilizable depends on the property called the decentralized fixed modes. In a system (centrally), fixed modes are the uncontrollable and unobservable modes of the system. In a decentralized control system, any uncontrollable and unobservable eigenvalue with respect to a centralized control is

also a fixed mode with respect to decentralized control [Davison and Özgüner, 1983]. Decentralized fixed modes includes any modes of the system which are not controllable and observable in the centralized sense, and those uncontrollable and unobservable modes that arise due to the decentralization of the system. If there are uncontrollable and unobservable modes for an overall system (centrally) which are unstable, then the system itself have to be change. Thus, in this study, it is assumed that the overall system is controllable and observable. It also assumed that if there is any decentralized fixed mode in the system due to decentralization, the decentralized fixed mode lies in the open left half part of the complex plane. In other words, it is assumed that there is no unstable decentralized fixed mode in the system.

In assumption (ii), the continuous functions  $H_i(X, \xi, t)$  and  $E_i(X, \xi, t)$  exist if and only if the following rank conditions are satisfied :

$$\text{rank} [ B_i ] = \text{rank} [ B_i, \Delta A_i(X, \xi, t) ] \quad (4.22)$$

$$\text{rank} [ B_i ] = \text{rank} [ B_i, \Delta B_i(X, \xi, t) ] \quad (4.23)$$

That is, if and only if each column of the matrices  $\Delta A_i(X, \xi, t)$  and  $\Delta B_i(X, \xi, t)$  is a linear combination of the columns of the matrix  $B_i$ . Equations (4.17) and (4.18) or the rank conditions (4.21) and (4.22) are essentially related to the structure of the matrices  $B_i$ ,  $\Delta A_i(X, \xi, t)$ , and  $\Delta B_i(X, \xi, t)$ , and not to the values of their elements. These conditions impose constraints on the structure of the system matrix uncertainty  $\Delta A_i(*)$ , and the input matrix uncertainty  $\Delta B_i(*)$  in that, the uncertainties should lie within the range space of the input matrix  $B_i$ . This assumption is needed so that the control,  $U_i(*)$ , which enters the  $i$ th subsystem through  $B_i$ , can overcome or compensate the possible uncertainty in the subsystem. If these assumptions are not satisfied, one cannot compensate for the uncertainties even if the uncertainties are known [Chen, 1987b]. In view of equations (3.21), these conditions are always satisfied for a robot manipulator if its state space description is derived based on the joint angles, velocities and acceleration as the state variables.



The condition  $\|E_i(*)\| < 1$  in assumption (ii) is needed to assure that a given control acts in the desired 'direction'. This condition also imposes a limit to the uncertainty present in the  $i$ th subsystem input matrix.

Assumption (iii) is required to ensure that the  $i$ th free decoupled subsystem state tracks the  $i$ th desired state trajectory  $X_{d,i}(t)$  asymptotically.

For each subsystem, let the state tracking error between the actual and the desired states be defined as  $Z_i(t)$ , that is,

$$Z_i(t) = X_i(t) - X_{d,i}(t) \quad (4.24)$$

and, 
$$Z_i(t_o) = X_i(t_o) - X_{d,i}(t_o) = Z_{i,o} . \quad (4.25)$$

Using (4.12) and (4.20), the error state equation for the  $i$ th subsystem can be obtained as follows :

$$\begin{aligned} \dot{Z}_i(t) = & A_i Z_i(t) + \Delta A_i(X, \xi, t) X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) + \\ & \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t) - B_i \Omega_i(t) , \end{aligned} \quad (4.26)$$

and the error state equation for the overall system may be obtained as follows :

$$\dot{Z}(t) = AZ(t) + A(X, \xi, t)X(t) - AX(t) + B(X, \xi, t)U(t) - B\Omega(t), \quad (4.27)$$

where

$$A = \text{diag} [ A_1, A_2, \dots, A_N ]$$

$$B = \text{diag} [ B_1, B_2, \dots, B_N ]$$

$$Z(t) = [ Z_1^T(t), Z_2^T(t), \dots, Z_N^T(t) ]^T$$

$$\Omega(t) = [ \Omega_1(t), \Omega_2(t), \dots, \Omega_N(t) ]^T.$$

The objective is then to design  $U_i(t)$  such that for any uncertainties, interconnection functions, and initial conditions, the overall error system (eqn. (4.27)) with the decentralized control law

$$U(t) = [ U_1(t), U_2(t), \dots, U_N(t) ]^T, \quad (4.28)$$

is practically stable. Hence, in the error space, the robot manipulator tracking problem has become the problem of stabilizing the error system (equation 4.27).

In the following section, a decentralized tracking controller based on local approach is presented. The vector norms are Euclidean, and the matrix norms are the corresponding induced one, that is,  $\|A\| = \sqrt{\lambda_{\max}(A^T A)}$ , where  $\lambda_{\max}(A)$  denotes the maximum eigenvalue of the matrix  $A$ .

#### 4.4 DECENTRALIZED NONLINEAR TRACKING CONTROLLER DESIGN – LOCAL APPROACH

In this section, a nonlinear decentralized local tracking controller for a robot manipulator system is proposed. Each local controller is designed based only on the local states of the corresponding subsystem.

From assumptions (ii) and (iii), with the arguments  $(X, \xi, t)$  has been dropped for convenience, equation (4.26) becomes :

$$\begin{aligned} \dot{Z}_i(t) = & A_i Z_i(t) + B_i \left\{ H_i X_i(t) + [I_3 + E_i] U_i(t) - \Omega_i(t) \right\} + \\ & \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij} X_j(t) . \end{aligned} \quad (4.29)$$

For each subsystem, let the decentralized local controller be of the form :

$$U_i(t) \triangleq U_{Li}(t) = \tilde{U}_i(Z_i, \xi, t) + \Phi_{Li}(Z_i, \xi, t) . \quad (4.30)$$

$$\tilde{U}_i(Z_i, \xi, t) = K_i Z_i(t) + \Omega_i(t), \quad (4.31)$$

where

$$\Phi_{L_i}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_{L_i}(Z_i, \xi, t)}{\|\mu_{L_i}(Z_i, \xi, t)\|} \rho_{L_i}(Z_i, \xi, t) & \text{if } \|\mu_{L_i}(Z_i, \xi, t)\| > \epsilon_i \\ \|\Phi_{L_i}(Z_i, \xi, t)\| \leq \rho_{L_i}(Z_i, \xi, t) & \text{if } \|\mu_{L_i}(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (4.32)$$

$$\mu_{L_i}(Z_i, \xi, t) = B_i^T P_i Z_i(t) \rho_{L_i}(Z_i, \xi, t) \quad (4.33)$$

$$\rho_{L_i}(Z_i, \xi, t) \triangleq \max_{\Delta b_i^j \in \mathcal{Y}} \left[ 1 - \|E_i(\Delta b_i^j)\| \right]^{-1} \left( \max_{\Delta a_{ij}^j \in \mathcal{R}} \|H_i(\Delta a_{ij}^j) X_i(t)\| + \max_{\Delta b_i^j \in \mathcal{Y}} \|E_i(\Delta b_i^j) \tilde{U}_i(Z_i, \xi, t)\| \right), \quad (4.34)$$

and  $\epsilon_i$  in equation (4.32) is a prescribed positive constant. The constant gain matrix  $K_i$  is chosen such that the closed loop system matrix

$$\tilde{A}_i \triangleq A_i + B_i K_i \quad (4.35)$$

is asymptotically stable.  $P_i$  is the solution of the matrix Lyapunov equation

$$P_i \tilde{A}_i + \tilde{A}_i^T P_i = -Q_i, \quad (4.36)$$

for a given positive definite symmetric matrix  $Q_i$ . The structure of the decentralized local controller is as illustrated in Figure 4.1.

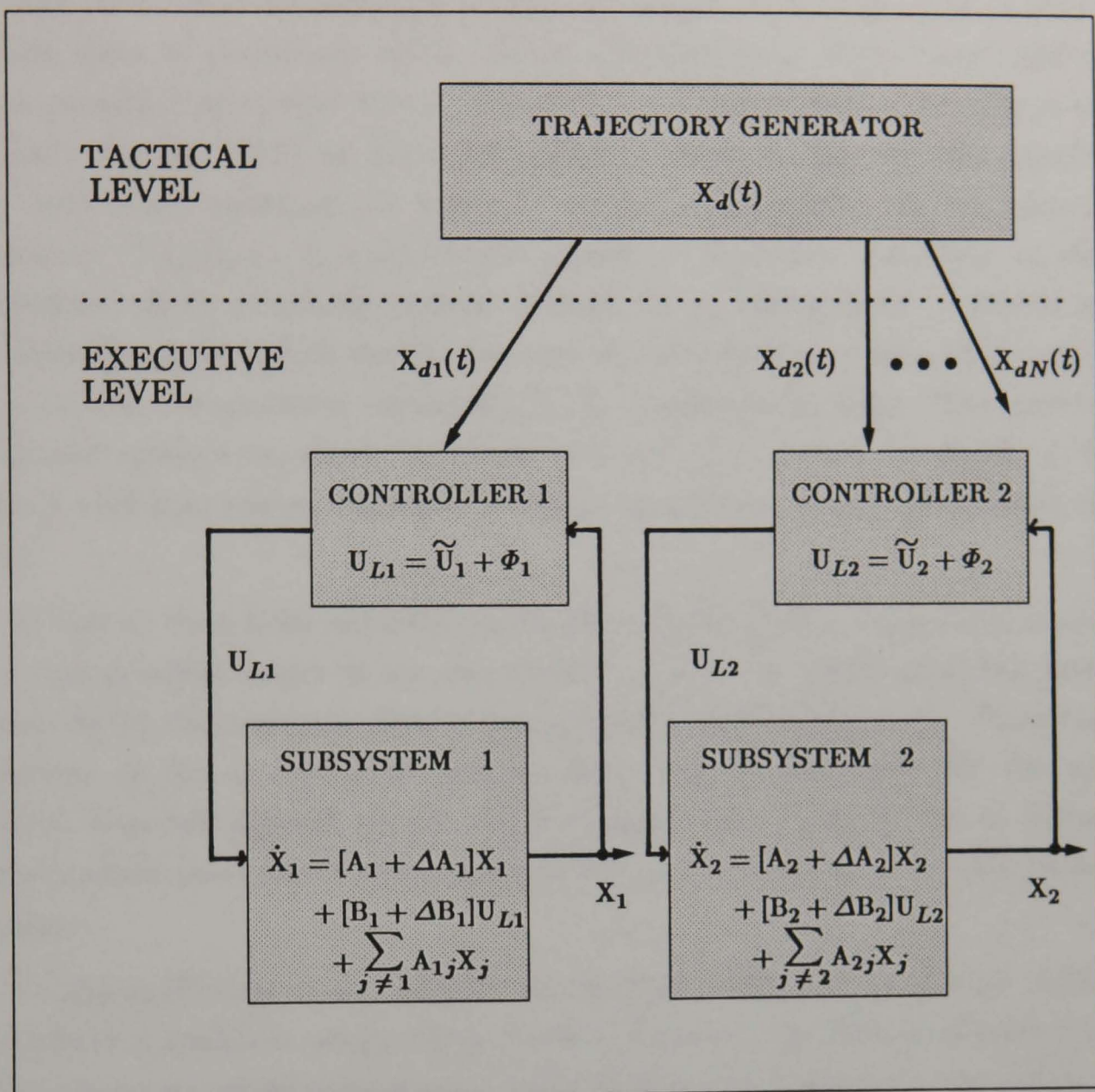


FIGURE 4.1 : Decentralized Control Of Robot Manipulator  
– Local Approach

As shown in equation (4.30), the controller is made up of a linear part and a nonlinear part. The linear part,  $\tilde{U}_i(Z_i, \xi, t)$ , is designed based on the  $i$ th decoupled nominal subsystem (without uncertainties), such that the  $i$ th isolated free nominal subsystem is stable and tracks the desired trajectory asymptotically.

The structure of the nonlinear component,  $\Phi_{Li}(Z_i, \xi, t)$ , as given in (4.32) is similar to those of [Leitmann, 1981; Corless and Leitmann, 1981; Chen, 1987a; 1988] as presented in section 4.2.1. It is of the saturation type. Its aim is to compensate for the effect of the uncertainties present in the  $i$ th subsystem's system and input matrices in order to reduce their effect on the system performance.  $\Phi_{Li}(Z_i, \xi, t)$  approximates closely a 'min-max' controller in the neighbourhood of the switching surface defined by  $\epsilon_i$ , where  $\epsilon_i$  can be viewed as a thin boundary layer which decides the size of the switching region, that is the region at which the nonlinear control  $\Phi_{Li}(Z_i, \xi, t)$  switches its form. The control action is continuous everywhere. If however,  $\epsilon_i = 0$ , then the control  $\Phi_{Li}(Z_i, \xi, t)$  becomes a switching control (min-max) which is discontinuous and unrealizable in practice.

It can be seen from equation (4.34) that  $\rho_{Li}(Z_i, \xi, t)$  is formulated based only on the possible bound of the uncertainty. Thus the uncertainty has been compensated by the nonlinear control component through  $\rho_{Li}(Z_i, \xi, t)$ . From the construction of the controller, it can be seen that the controller for the  $i$ th subsystem does not depend on the neighbouring states,  $X_j(t)$  (for  $j \neq i$ ). Thus, the decentralized nonlinear tracking controller  $U_{Li}(t)$  is designed based only on its local states.

Unlike in [Corless et.al., 1984; Shoureshi, 1987; 1990], the control law (4.30) does not have a nonlinear compensator which is based on the inverse dynamics of the manipulator to cancel the nonlinear and coupling terms present in the system. Furthermore, the controller is decentralized in structure. Hence, the computational complexities of the controller are greatly reduced.

#### Theorem 4.1

Subject to Assumptions (i) to (iv), the decentralized tracking controller (4.30) will render the interconnected robot manipulator system (4.12) globally practically stable and tracks the desired state trajectory (4.16) to within the neighbourhood of the ultimate boundedness set  $\mathfrak{X}(\underline{\kappa})$  (defined below) if the test matrix  $\mathbb{T}_L = [\mathbb{T}_L]_{N \times N}$ , where

$$\mathbf{T}_{Lij} = \begin{cases} \lambda_{\min}(Q_i) & \text{for } i = j \\ -2 \lambda_{\max}(P_i) \bar{g}_{ij} & \text{if } i \neq j \end{cases} \quad (4.37)$$

is a positive definite matrix, that is, if the successive principle minors of the test matrix are all positive.

### Proof

Substituting equation (4.30) into (4.29), gives the closed loop equation for the  $i$ th subsystem as (the arguments have been dropped for simplicity) :

$$\begin{aligned} \dot{Z}_i &= \tilde{A}_i Z_i + B_i \left\{ H_i X_i + [I_3 + E_i] \Phi_{Li} + E_i \tilde{U}_i \right\} + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij} X_j, \\ &= \tilde{A}_i Z_i + B_i \left\{ \Phi_{Li} + e_{Li} \right\} + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij} X_j, \end{aligned} \quad (4.38)$$

where

$$e_{Li} = H_i X_i + E_i \Phi_{Li} + E_i \tilde{U}_i. \quad (4.39)$$

Let the Lyapunov function for  $i$ th subsystem be :

$$\mathcal{L}_i(Z_i, t) = Z_i^T P_i Z_i, \quad (4.40)$$

and the composite Lyapunov function be :

$$\mathcal{L}(Z, t) = \sum_i^N L_i(Z_i, t) = \sum_i^N Z_i^T P_i Z_i. \quad (4.41)$$

The Lyapunov function  $\mathcal{L}(Z, t)$  is positive definite since  $P$  is positive definite. The

derivative of the Lyapunov function with respect to time  $t$  can be obtained as follows :

$$\dot{\mathcal{L}}(Z, t) = \sum_{i=1}^N 2 Z_i^T P_i \dot{Z}_i . \quad (4.42)$$

Substituting equation (4.38) into (4.42), gives :

$$\begin{aligned} \dot{\mathcal{L}}(Z, t) &= \sum_{i=1}^N 2 Z_i^T P_i \tilde{A}_i Z_i + \sum_{i=1}^N 2 Z_i^T P_i B_i \{ \Phi_{L_i} + e_{L_i} \} + \\ &\quad \sum_{i=1}^N 2 Z_i^T P_i \sum_{j \neq i}^N A_{ij} X_j \\ &= \sum_{i=1}^N 2 Z_i^T P_i \tilde{A}_i Z_i + \sum_{i=1}^N 2 (B_i^T P_i Z_i)^T \{ \Phi_{L_i} + e_{L_i} \} + \\ &\quad \sum_{i=1}^N 2 Z_i^T P_i \sum_{j \neq i}^N A_{ij} X_j . \end{aligned} \quad (4.43)$$

From Rayleigh's principle [Franklin, 1968] :

$$\lambda_{\min}(Q_i) \|Z_i\|^2 \leq Z_i^T Q_i Z_i \leq \lambda_{\max}(Q_i) \|Z_i\|^2 , \quad (4.44)$$

the first term on the right-hand-side (RHS) of equation (4.43) can be written as follows :

$$\begin{aligned} \sum_{i=1}^N 2 Z_i^T P_i \tilde{A}_i Z_i &= \sum_{i=1}^N Z_i^T [ P_i \tilde{A}_i + \tilde{A}_i^T P_i ] Z_i \\ &= - \sum_{i=1}^N Z_i^T Q_i Z_i \\ &\leq - \sum_{i=1}^N \lambda_{\min}(Q_i) \|Z_i\|^2 . \end{aligned} \quad (4.45)$$

From the fact that  $\|\Phi_{L_i}\| \leq \rho_{L_i}$  (equation 4.32) , implies,

$$\begin{aligned}
\| \mathbf{e}_{Li} \| &= \| \mathbf{H}_i \mathbf{X}_i + \mathbf{E}_i \boldsymbol{\Phi}_{Li} + \mathbf{E}_i \tilde{\mathbf{U}}_i \| \\
&\leq \| \mathbf{H}_i \mathbf{X}_i \| + \| \mathbf{E}_i \| \| \boldsymbol{\Phi}_{Li} \| + \| \mathbf{E}_i \| \| \tilde{\mathbf{U}}_i \| \\
&\leq \rho_{Li} \quad .
\end{aligned} \tag{4.46}$$

In deriving equation (4.46), equation (4.34) has been utilized. For the second term on the RHS of equation (4.43) :

i. If  $\mu_{Li} > \epsilon_i$ , implies

$$\boldsymbol{\Phi}_{Li} = - \frac{\mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i}{\| \mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i \|} \rho_{Li}, \tag{4.47}$$

which gives

$$\begin{aligned}
\sum_i^N 2(\mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i)^T \{ \boldsymbol{\Phi}_{Li} + \mathbf{e}_{Li} \} &\leq \sum_i^N 2 \| \mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i \| \| \boldsymbol{\Phi}_{Li} + \mathbf{e}_{Li} \| \\
&\leq \sum_i^N 2 \| \mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i \| \{ \| \boldsymbol{\Phi}_{Li} \| + \| \mathbf{e}_{Li} \| \} \\
&\leq - \sum_i^N 2 \| \mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i \| \rho_{Li} + \sum_i^N 2 \| \mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i \| \rho_{Li} \\
&= 0 \quad ;
\end{aligned} \tag{4.48}$$

ii. If  $\mu_{Li} \leq \epsilon_i$ , hence,

$$\| \mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i \| \rho_{Li} \leq \epsilon_i, \tag{4.49}$$

then,

$$\sum_i^N 2(\mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i)^T \{ \boldsymbol{\Phi}_{Li} + \mathbf{e}_{Li} \} \leq \sum_i^N 2 \| \mathbf{B}_i^T \mathbf{P}_i \mathbf{Z}_i \| \| \boldsymbol{\Phi}_{Li} + \mathbf{e}_{Li} \|$$



$$\begin{aligned}
&\leq \sum_i^N 2 \|B_i^T P_i Z_i\| \|\Phi_{L_i}\| + \sum_i^N 2 \|B_i^T P_i Z_i\| \|e_{L_i}\| \\
&\leq \sum_i^N 2 \|B_i^T P_i Z_i\| \rho_{L_i} + \sum_i^N 2 \|B_i^T P_i Z_i\| \rho_{L_i} \\
&\leq \sum_i^N 4 \|B_i^T P_i Z_i\| \rho_{L_i} \\
&\leq \sum_i^N 4 \epsilon_i .
\end{aligned} \tag{4.50}$$

Now, consider the error vector

$$Z_j = X_j - X_{jd} \tag{4.51}$$

Then,  $\|Z_j\| = \|X_j - X_{jd}\|$ . (4.52)

But, from Pollard et. al. [1986] ,

$$\|X_j\| - \|X_{jd}\| \leq \|X_j - X_{jd}\| \leq \|X_j\| + \|X_{jd}\|, \tag{4.53}$$

hence,  $\|X_j\| \leq \|Z_j\| + \|X_{jd}\|$ . (4.54)

Then, the last term on the RHS of equation (4.43) can be written in the following form :

$$\begin{aligned}
\sum_i^N 2Z_i^T P_i \sum_{j \neq i}^N A_{ij} X_j &\leq \sum_i^N 2\lambda_{\max}(P_i) \|Z_i\| \sum_{j \neq i}^N \|A_{ij}\| \|X_j\| \\
&\leq \sum_i^N \sum_{j \neq i}^N 2\lambda_{\max}(P_i) \|Z_i\| \bar{g}_{ij} \|X_j\| \\
&\leq \sum_i^N \sum_{j \neq i}^N 2\lambda_{\max}(P_i) \|Z_i\| \left\{ \bar{g}_{ij} (\|Z_j\| + \|X_{jd}\|) \right\} \\
&= \sum_i^N \sum_{j \neq i}^N 2\lambda_{\max}(P_i) \left\{ \bar{g}_{ij} \|Z_i\| \|Z_j\| + \bar{g}_{ij} \|X_{jd}\| \|Z_i\| \right\}.
\end{aligned} \tag{4.55}$$

In deriving equation (4.55), equation (4.54) and assumption (iv) have been utilized. From equations (4.45), (4.48), (4.50) and (4.55), the Lyapunov derivative (4.43) along the solution of the error system (4.38) is given as follows :

$$\begin{aligned} \dot{\mathbf{L}}(\mathbf{Z}, t) \leq & - \sum_i^N \lambda_{\min}(\mathbf{Q}_i) \|\mathbf{Z}_i\|^2 + \sum_i^N 4\epsilon_i + \\ & \sum_i^N \sum_{j \neq i}^N 2\lambda_{\max}(\mathbf{P}_i) \left\{ \bar{g}_{ij} \|\mathbf{Z}_i\| \|\mathbf{Z}_j\| + \bar{g}_{ij} \|\mathbf{X}_{jd}\| \|\mathbf{Z}_i\| \right\}. \end{aligned} \quad (4.56)$$

Let, 
$$\tilde{\mathbf{Z}}^T = [ \|\mathbf{Z}_1\|, \|\mathbf{Z}_2\|, \dots, \|\mathbf{Z}_N\| ], \quad (4.57)$$

hence, 
$$\tilde{\mathbf{Z}}^T \tilde{\mathbf{Z}} = \|\tilde{\mathbf{Z}}\|^2 = \|\mathbf{Z}\|^2. \quad (4.58)$$

Then, the derivative of the Lyapunov equation (4.56) can be rewritten as follows :

$$\dot{\mathbf{L}}(\mathbf{Z}, t) \leq - \tilde{\mathbf{Z}}^T \mathbb{T}_L \tilde{\mathbf{Z}} + \mathbb{V} \tilde{\mathbf{Z}} + \sum_i^N 4 \epsilon_i \quad (4.59)$$

where,  $\mathbb{T}_L$  is an  $N \times N$  test matrix whose elements are as given in equation (4.37), and,  $\mathbb{V}$  is an  $N \times N$  matrix whose elements are as follows :

$$\mathbb{V}_{ij} = \begin{cases} \sum_{j \neq i}^N 2 \lambda_{\max}(\mathbf{P}_i) \bar{g}_{ij} \|\mathbf{X}_{jd}\| & \text{for } i = j \\ 0 & \text{if } i \neq j \end{cases}. \quad (4.60)$$

Equation (4.59) can be rewritten as :

$$\dot{\mathbf{L}}(\mathbf{Z}, t) \leq - \lambda_{\min}(\mathbb{T}_L) \|\tilde{\mathbf{Z}}\|^2 + \|\mathbb{V}\| \|\tilde{\mathbf{Z}}\| + \sum_i^N 4 \epsilon_i. \quad (4.61)$$

It can be seen that (4.61) is of the form of equation (4.6) in section 4.2.1. Thus, if  $\lambda_{\min}(\mathbb{T}_L) > 0$ ,  $\dot{\mathbf{L}}(\mathbf{Z}, t) < 0$  for all  $t$  and  $\tilde{\mathbf{Z}} \in \mathfrak{B}^c(\eta_L)$ , where  $\mathfrak{B}^c(\eta_L)$  is the

complement of the closed ball  $\mathfrak{B}(\eta_L)$ , centred at  $\tilde{Z}=0$  with radius

$$\eta_L = \frac{\|\mathbf{V}\|}{2\lambda_{\min}(\mathbb{T}_L)} + \left[ \left( \frac{\|\mathbf{V}\|}{2\lambda_{\min}(\mathbb{T}_L)} \right)^2 + \frac{\sum_i^N \epsilon_i}{\lambda_{\min}(\mathbb{T}_L)} \right]^{\frac{1}{2}}, \quad (4.62)$$

and the error system (4.38) with the nonlinear decentralized controller (4.30) and corresponding to an initial condition  $(X_0, t_0)$ , is uniform ultimate boundedness with respect to the set  $\mathfrak{X}(\underline{\kappa})$ , where

$$\mathfrak{X}(\underline{\kappa}) \triangleq \left\{ \tilde{Z} \in \mathbb{R}^N \mid \tilde{Z}^T \mathbf{P} \tilde{Z} < \underline{\kappa} \right\}, \quad (4.63)$$

$$\underline{\kappa} = \lambda_{\max}(\mathbf{P}) \eta_L^2, \quad (4.64)$$

$$\mathbf{P} = \text{diag} [ \mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_N ]. \quad (4.65)$$

This conclude the proof.  $\square$

The condition in terms of the positive definiteness of the test matrix  $\mathbb{T}_L$  as given by equation (4.37) is a sufficient condition for the practical stability of the system (but not necessary). It should be noted that Lyapunov stability theory is always known to be conservative in predicting stability of a system. If the condition is satisfied then the system is practically stable with respect to the stability region. If the test is not successful, one cannot conclude whether the subsystem is stable or not; the system may be stable even when the test is not positive. Another important point that should be mentioned concerning the test matrix  $\mathbb{T}_L$  is that, the weaker the interconnections, that is, the smaller the terms  $\bar{g}_{ij}$ , for  $i \neq j$ , the easier it is to satisfy the sufficient condition.

Inspection of (4.62) shows that the size of the set  $\mathfrak{X}(\underline{\kappa})$ , and hence the tracking error, depends both on  $\epsilon_i$  and the bound of the interconnection matrices  $\bar{g}_{ij}$ . For small values of  $\epsilon_i$ , the radius of the uniform ultimate boundedness set decreases and hence, the tracking precision increases. Thus, the errors between the responses of the robotic system and the reference trajectories can be made arbitrarily small. In other words, the robot manipulator can be made to track the reference trajectory within an arbitrarily small neighbourhood of  $X_d(t)$ .

The decentralized control law (4.30) is simple and easy to compute because it is designed based on a set of decomposed robot manipulator models, rather than the overall system. Moreover, it is physically realizable and easy to implement as the exchange of any information between subsystems are not required.

The performance of the proposed decentralized control law is evaluated by means of a computer simulation study which will be presented in the following section.

## 4.5 SIMULATION AND RESULTS

A simulation study has been carried out to investigate the performance of the proposed controller on a three dof robot manipulator actuated by armature controlled DC motors.

The complete dynamic model of the robot manipulator has been obtained in Chapter 2. Equation (2.75) which represents the complete dynamics of the robot manipulator is highly nonlinear and coupled, taking into account the contributions of the actuator dynamics, as well as the inertias, the Coriolis forces, the centrifugal forces and the gravitational forces present in the mechanical part of the robot arm. It is this equation (equation 2.75) that was used in the simulation to represent a real plant (three dof robot manipulator) without any approximation and simplification of the highly nonlinear and coupled system.

In this simulation work, the Runge-Kutta-Butcher (James et.al, 1985) numerical integration method has been used to solve the robot nonlinear differential equations. The method was used because it is one of the best methods

in finding a solution to a differential equation in terms of accuracy and minimum computing time (James, et. al., 1985). Computer programs in FORTRAN have been developed for simulating the dynamic behaviour of the robot manipulator with the controller. Double precision arithmetic and a sampling period of 10ms have been used for the simulations throughout this study.

In the simulations, as in any actual robot manipulator application, it is necessary to have a preplanned trajectory for the robot manipulator to follow so that it can move efficiently with or without any obstacle in the work space.

#### 4.5.1 Trajectory Generation

At the tactical level of the overall robot control hierarchy as described in Chapter 1, the desired trajectory for each joint angle (position), velocity, and acceleration of the robot manipulator is generated using a trajectory generator.

A trajectory is defined as a time history of position, velocity and acceleration of each joint (each dof) of the robot manipulator. Logically, a smooth continuous trajectory is preferable than a jerky discontinuous one. This requires that the first and the second derivative of the trajectory must exist to guarantee the continuity of the trajectory. Suppose that it is desired for the  $i$ th robot joint angle  $\theta_i(t)$  to move from a given initial position  $\theta_i(0)$  to a desired final destination of  $\theta_i(\tau)$  in time interval  $\tau$  seconds, starting and ending with zero velocity, that is,  $\dot{\theta}_i(0) = \dot{\theta}_i(\tau) = 0$  rad/s. One way to generate a smooth joint space trajectory with the specified conditions for the joint angle is by using a cycloidal function as follows :

$$\theta_{di}(t) = \begin{cases} \theta_i(0) + \frac{\Delta_i}{2\pi} \left[ \frac{2\pi t}{\tau} - \sin\left(\frac{2\pi t}{\tau}\right) \right], & 0 \leq t \leq \tau \\ \theta_i(\tau), & \tau \leq t \end{cases} \quad (4.66)$$

where

$$\Delta_i = \theta_i(\tau) - \theta_i(0) \quad , \quad i = 1, 2, 3.$$

The subscript  $d$  means the desired trajectory for the joint to track. The cycloid has a smooth 'bell-shaped' velocity profile and a 'sinusoidal' acceleration profile which can be obtained by differentiating equation (4.67) once and twice, respectively, with respect to time  $t$ :

$$\dot{\theta}_{di}(t) = \begin{cases} \frac{\Delta_i}{\tau} \left[ 1 - \cos \left( \frac{2\pi t}{\tau} \right) \right], & 0 \leq t \leq \tau \\ 0, & \tau \leq t \end{cases} \quad (4.67)$$

$$\ddot{\theta}_{di}(t) = \begin{cases} \frac{2\pi\Delta_i}{\tau^2} \sin \left( \frac{2\pi t}{\tau} \right), & 0 \leq t \leq \tau \\ 0, & \tau \leq t \end{cases} \quad (4.68)$$

The cycloid also has a smooth jerk profile  $\dddot{\theta}_{di}(t)$  which can be obtained by differentiating equation (4.68) once with respect to time.

Figure 4.2 illustrates the joint angular trajectory profile for each joint of the three dof robot manipulator starting from an initial position of  $[\theta_1(0), \theta_2(0), \theta_3(0)]^T = [-0.8, -1.5, -0.5]^T$  radians, to a desired final position of  $[\theta_1(\tau), \theta_2(\tau), \theta_3(\tau)]^T = [1.0, 0.2, 1.2]^T$  radians in time  $\tau = 2$  seconds. Figures 4.3 and 4.4 show the corresponding velocity and acceleration profiles for each joint of the robot manipulator, respectively. It can be seen that the trajectories are smooth and continuous.

In the simulations, care has been taken in selecting the time  $\tau$  for the joint to move from a given initial angular position to a desired final position to ensure that the maximum velocity profile generated does not exceed the specified maximum velocity of the robot manipulator. From equations (4.68) and (4.69), the time  $\tau$  needed for the joint to move from a given initial position to a final position should be chosen such that

### Desired Joint Angle Trajectory

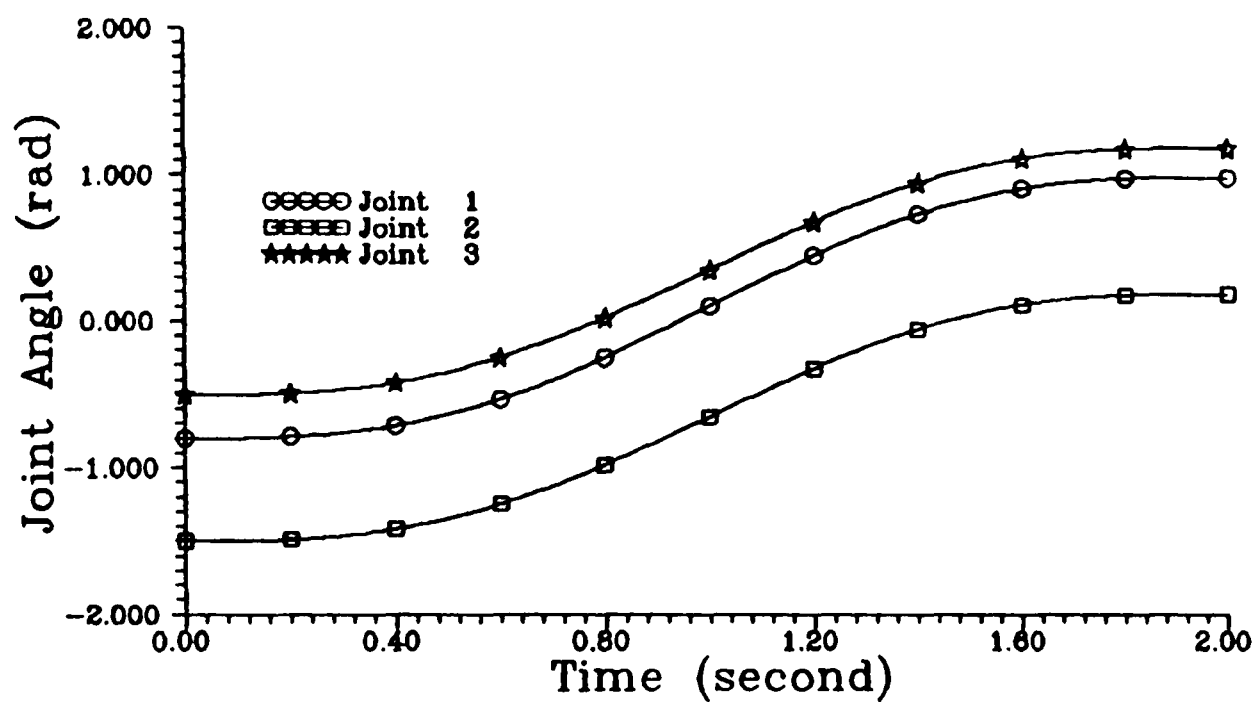


FIGURE 4.2 : Desired Joint Position profile

### Desired Joint Velocity Trajectory

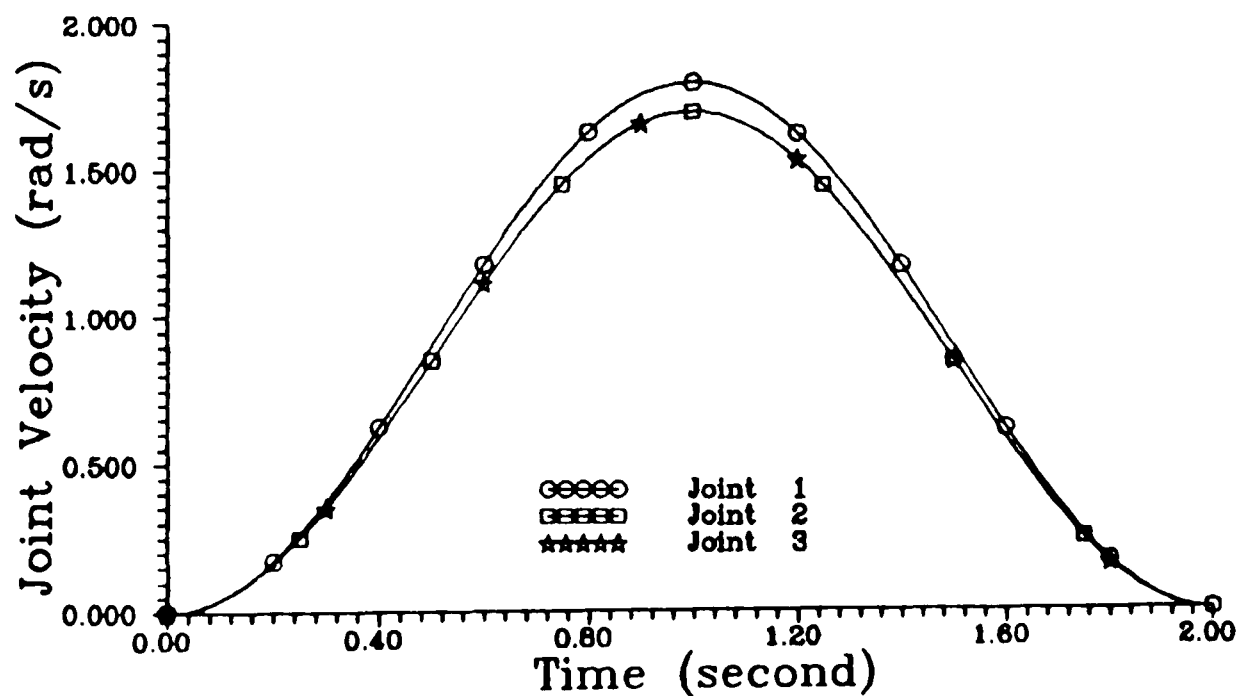


FIGURE 4.3 : Desired Joint Velocity Profile

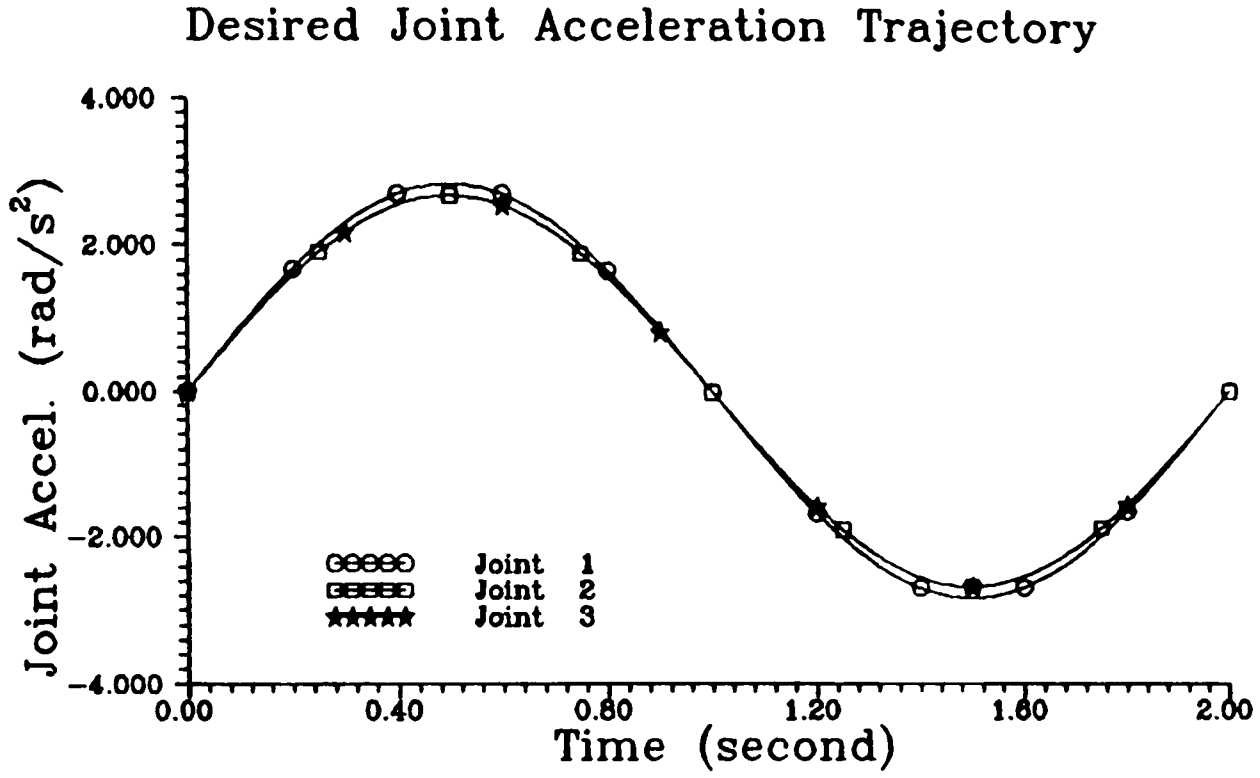


FIGURE 4.4 : Desired Joint Acceleration Profile

$$\tau \leq \frac{2\Delta_i}{\dot{\theta}_m} , \quad (4.69)$$

where  $\dot{\theta}_m$  is the maximum velocity specified for the robot manipulator.

Throughout this research, each joint of the robot manipulator is simulated to track the above type of trajectories.



#### 4.5.2 Simulation Using Independent Joint Linear Control

Before the proposed decentralized nonlinear controller is applied to the robot manipulator, a simulation study was carried out to evaluate the performance of the robot manipulator using an independent joint control method which is normally used in a real industrial robot. Here a linear state feedback controller was used for each joint. In designing the controller, the dynamics of the mechanical linkage has been completely ignored. Each joint of the robot manipulator was treated as an independent servomechanism problem represented by the actuator state equation as follows :

$$\dot{X}_i(t) = A_{Bi} X_i(t) + B_{Bi} U_i(t) . \quad (4.70)$$

The above equation is obtained by ignoring the last two terms on the right hand side of equation (2.26). The elements of the matrices  $A_{Bi}$  and  $B_{Bi}$  are as in equation (2.27). Based on the actuator's parameters value as tabulated in Table 2-2 in Chapter 2, the matrices  $A_{Bi}$  and  $B_{Bi}$  are as follow :

$$A_{Bi} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -90.6955 & -10.8968 \end{bmatrix}, \quad i = 1, 2, 3, \quad (4.71)$$

$$B_{B1} = \begin{bmatrix} 0 \\ 0 \\ 0.7217 \end{bmatrix}, \quad B_{B2} = B_{B3} = \begin{bmatrix} 0 \\ 0 \\ 0.6415 \end{bmatrix}.$$

The following linear state feedback controller was employed for each subsystem :

$$U_i(t) = K_i Z_i(t) + \Omega_{mi}(t) , \quad (4.72)$$

where  $K_i$  is the  $1 \times 3$  linear state feedback gain matrix,  $Z_i(t)$  is the state error vector similar to equation (4.23), and  $\Omega_{mi}(t)$  is the control component to ensure asymptotic tracking of the  $i$ th open-loop decoupled linear subsystem (4.71) with respect to the desired state trajectory vector  $X_{di}(t) = [\theta_{di}(t), \dot{\theta}_{di}(t), \ddot{\theta}_{di}(t)]^T$ .

The following  $\Omega_{mi}(t)$  functions have been used in the simulation :

$$\begin{aligned} \Omega_{m1}(t) &= 125.6744 \dot{\theta}_{d1}(t) + 15.2241 \ddot{\theta}_{d1}(t) + 1.3857 \ddot{\theta}_{d1}(t) \\ \Omega_{m2}(t) &= 141.3837 \dot{\theta}_{d2}(t) + 17.1272 \ddot{\theta}_{d2}(t) + 1.5589 \ddot{\theta}_{d2}(t) \\ \Omega_{m3}(t) &= 141.3837 \dot{\theta}_{d3}(t) + 17.1272 \ddot{\theta}_{d3}(t) + 1.5589 \ddot{\theta}_{d3}(t) . \end{aligned} \quad (4.73)$$

For the following closed-loop poles of the subsystems :

$$\begin{aligned} \text{Subsystem 1} &: -0.4, -0.4, -5.0 \\ \text{Subsystem 2} &: -5.0, -5.0, -10.0 \\ \text{Subsystem 3} &: -0.3, -0.3, -5.0 \end{aligned}$$

the corresponding feedback gains have been computed as follow :

$$\begin{aligned} \text{Subsystem 1} &: K_1 = \begin{bmatrix} -1.1085 & 119.91 & 7.1873 \end{bmatrix} \\ \text{Subsystem 2} &: K_2 = \begin{bmatrix} -389.721 & -53.4767 & -14.0505 \end{bmatrix} \\ \text{Subsystem 3} &: K_3 = \begin{bmatrix} -0.7015 & 136.5668 & 8.3975 \end{bmatrix} . \end{aligned} \quad (4.74)$$

Four cases have been considered in the simulation study :

- a) the manipulator was required to track a slow trajectory without carrying any load
- b) the manipulator was required to track a fast trajectory without carrying any load
- c) the manipulator was required to track a slow trajectory while carrying a 20 Kg. load
- d) the manipulator was required to track a fast trajectory while carrying a 20 Kg. load.

For the slow trajectory, the robot manipulator was required to move from an initial position of  $\theta(0) = [ 0.0 , -1.5 , -0.2 ]^T$  radians to the final position of  $\theta(\tau) = [ 0.5 , -1.0 , -0.3 ]^T$  radians in  $\tau = 2$  seconds. The maximum velocity for this trajectory is 0.5 radian per second for all the joints. For the fast trajectory, the robot manipulator was required to move from an initial position of  $\theta(0) = [ -0.8 , -1.5 , -0.5 ]^T$  radians to the final position of  $\theta(\tau) = [ 1.0 , 0.2 , 1.2 ]^T$  radians in  $\tau = 2$  seconds. The maximum velocity for this trajectory is at 1.8 radians per second for joint 1, and 1.7 radians per second for joint 2 and 3.

Figure 4.5 illustrates the position tracking error for each joint for case (a) and (b), while Figure 4.6 shows the position tracking error for case (c) and (d). From the graphs, it can be observed that the tracking errors are significant when the manipulator was carrying a load compared to without load. It can be seen from Figure 4.5 that the error for each joint angle is significant for the case when the manipulator was required to track a fast trajectory. Similar results can also be observed from Figure 4.6 where the manipulator was required to carry a 20 Kg. load. The increase in the load or velocity or both may also cause instability as shown by the simulation results for joint 3 of the manipulator in Figure 4.6 .

The simulation results confirm that the independent linear joint controller is not robust to load variations and is only suitable for slow manipulator motion. For fast manipulator motion, the influence of the dynamics of the mechanical part of the manipulator is strong due to increase in the velocity dependent

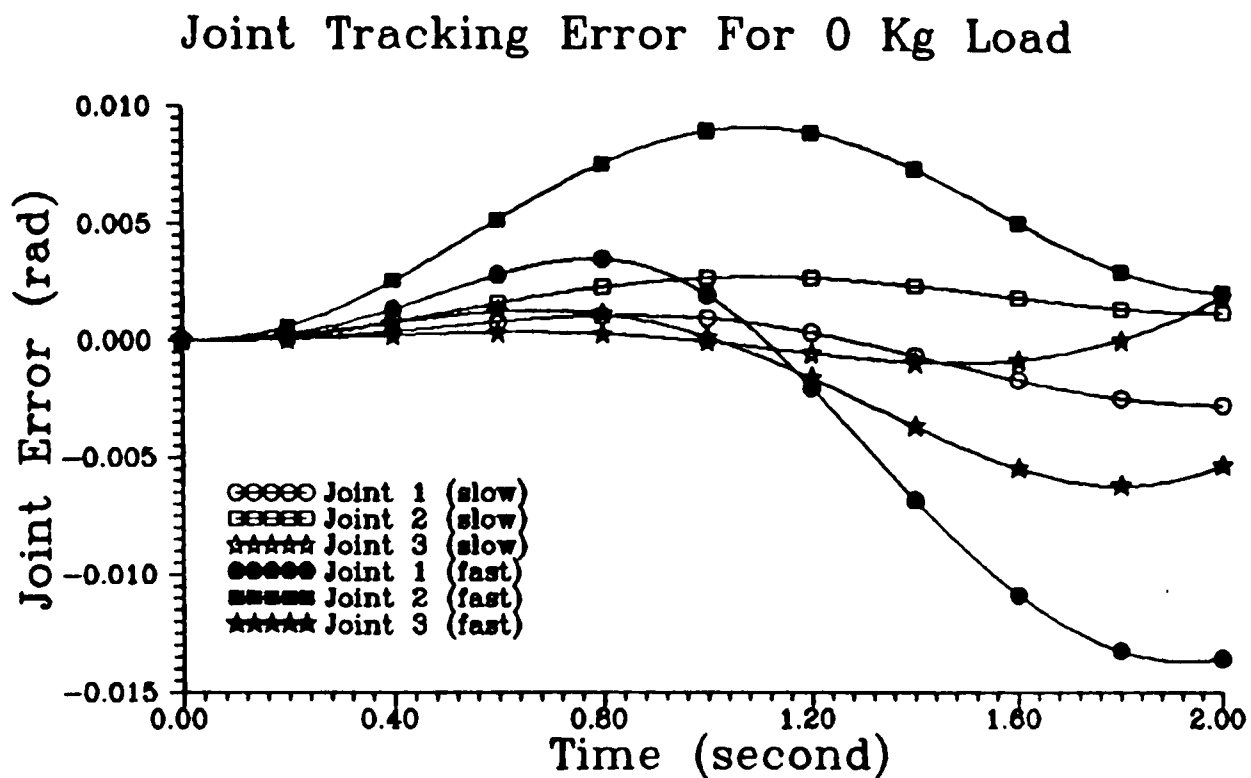


FIGURE 4.5 : Joint Tracking Error Without Load Using Linear Control Method

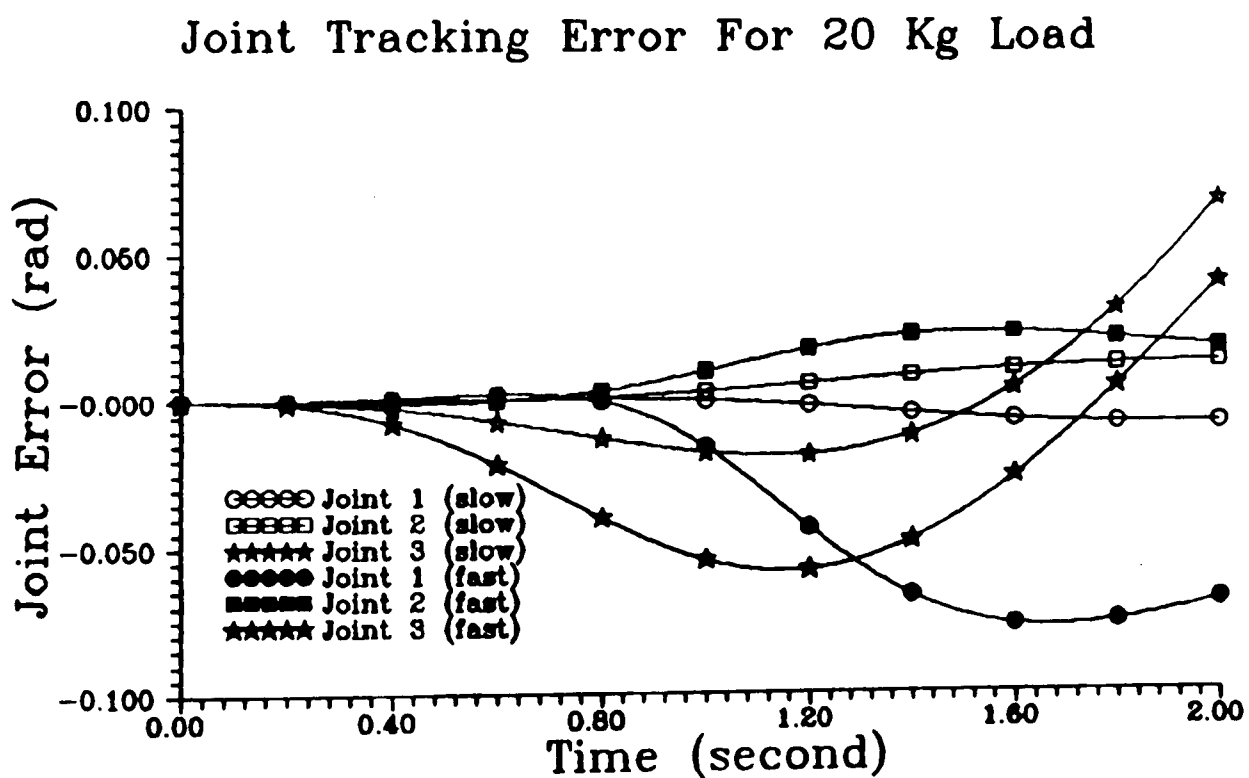


FIGURE 4.6 : Joint Tracking Error For 20 Kg. Load Using Linear Controller

component in the mechanical linkage dynamics such as Coriolis forces terms. The increase in the load carried by the manipulator also increases among others the gravitational loading terms in the mechanical linkage dynamics. The static linear controller with a particular fixed set of feedback gains does not have the capability to compensate these sudden changes in the overall robot manipulator system, hence resulting in a large tracking error and may cause instability of the system as shown in the simulations. In designing a controller for a robot manipulator which is required to move payloads of different masses from one point to another, it is more desirable to have a single controller for each joint which works for a range of payload masses and trajectories than to change the controller or adjust some controller parameters for each task.

#### 4.5.3 Simulation Using Decentralized Nonlinear Local Control Method

In order to apply the proposed decentralized control strategy, the robot dynamic model was decomposed and reduced into a set of three interconnected uncertain subsystems in input decentralized form as outlined in section 3.4 of Chapter 3. From equation (3.27), the dynamic equations of the subsystems are as follows :

$$\begin{aligned}
 \dot{X}_1(t) &= [A_1 + \Delta A_1(X, \xi, t)] X_1(t) + [B_1 + \Delta B_1(X, \xi, t)] U_1(t) + \\
 &\quad A_{12}(X, \xi, t) X_2(t) + A_{13}(X, \xi, t) X_3(t) \\
 \dot{X}_2(t) &= [A_2 + \Delta A_2(X, \xi, t)] X_2(t) + [B_2 + \Delta B_2(X, \xi, t)] U_2(t) + \\
 &\quad A_{21}(X, \xi, t) X_1(t) + A_{23}(X, \xi, t) X_3(t) \\
 \dot{X}_3(t) &= [A_3 + \Delta A_3(X, \xi, t)] X_3(t) + [B_3 + \Delta B_3(X, \xi, t)] U_3(t) + \\
 &\quad A_{31}(X, \xi, t) X_1(t) + A_{32}(X, \xi, t) X_2(t) .
 \end{aligned} \tag{4.75}$$

The non-zero elements of the uncertain matrices  $\Delta A_i(X, \xi, t)$  and  $\Delta B_i(X, \xi, t)$  for  $i=1, 2, 3$ , can be obtained by using equations (3.15), (3.16), (3.22), (3.23) and Appendix D. In view of equation (3.22), the non-zero elements of the interconnection matrices  $A_{ij}(X, \xi, t)$  for  $i \neq j$  are as given in the Appendix D. The elements of the matrices  $A_i$  and  $B_i$ , as well as the bounds on the elements of the matrices  $\Delta A_i(X, \xi, t)$ ,  $\Delta B_i(X, \xi, t)$  and  $A_{ij}(X, \xi, t)$  are as given by equation (3.26) in Chapter 3. Based on equation (4.76) and the known maximum bounds on the uncertainties in the system - the maximum bound on the uncertain elements of the system, input and interconnection matrices - the decentralized control law is designed to control the robot manipulator.

It should be emphasized here that the robot dynamics in the form of equation (4.76) is only used for the derivation of the decentralized nonlinear control law, which is done off-line.

For each subsystem, the control law was established according to equations (4.30) to (4.36). Based on equation (4.35), the linear feedback gains in equation (4.31) were synthesized for each subsystem closed-loop poles similar to those used in the previous section for comparison purposes. The following corresponding feedback gains for each subsystem were obtained :

$$\begin{aligned}
 \text{Subsystem 1} : \quad K_1 &= \begin{bmatrix} -1.145 & 119.7204 & 7.3775 \end{bmatrix} \\
 \text{Subsystem 2} : \quad K_2 &= \begin{bmatrix} -400.0975 & -58.9978 & -14.4766 \end{bmatrix} \\
 \text{Subsystem 3} : \quad K_3 &= \begin{bmatrix} -0.7088 & 136.5112 & 8.4698 \end{bmatrix} .
 \end{aligned} \tag{4.76}$$

Notice that the feedback gains are not much different from those in the previous section. The functions  $\Omega_i(t)$  in equation (4.31) have been determined as follows :

$$\begin{aligned}
\Omega_1(t) &= 125.6744 \dot{\theta}_{d1}(t) + 15.6788 \ddot{\theta}_{d1}(t) + 1.4313 \dddot{\theta}_{d1}(t) \\
\Omega_2(t) &= 141.0512 \dot{\theta}_{d2}(t) + 17.5313 \ddot{\theta}_{d2}(t) + 1.6004 \dddot{\theta}_{d2}(t) \\
\Omega_3(t) &= 141.3782 \dot{\theta}_{d3}(t) + 17.2894 \ddot{\theta}_{d3}(t) + 1.5751 \dddot{\theta}_{d3}(t) .
\end{aligned} \tag{4.77}$$

In solving the matrix Lyapunov equation (equation 4.36), the positive definite symmetrical matrix  $Q_i$  was taken to be a 3x3 identity matrix for each subsystem. The following solution of the Lyapunov equation has been computed :

$$\begin{aligned}
P_1 &= \begin{bmatrix} 3.4341 & 3.8378 & 0.625 \\ 3.8378 & 6.5294 & 1.0427 \\ 0.625 & 1.0427 & 0.266 \end{bmatrix} , \\
P_2 &= \begin{bmatrix} 15.3389 & 7.0444 & 0.002 \\ 7.0444 & 4.7073 & 0.0604 \\ 0.002 & 0.0604 & 0.02802 \end{bmatrix} , \\
P_3 &= \begin{bmatrix} 4.4444 & 6.443 & 1.1111 \\ 6.443 & 12.9873 & 2.2469 \\ 1.1111 & 2.2469 & 0.4905 \end{bmatrix} .
\end{aligned} \tag{4.78}$$

Next, the functions  $H_i(X, \xi, t)$  and  $E_i(X, \xi, t)$  are computed as follows :

$$\begin{aligned}
H_1(X, \xi, t) &= 1.4313 \begin{bmatrix} 0 & \Delta a_{32}^1 & \Delta a_{33}^1 \end{bmatrix} \\
H_2(X, \xi, t) &= 1.6004 \begin{bmatrix} \Delta a_{31}^2 & \Delta a_{32}^2 & \Delta a_{33}^2 \end{bmatrix} \\
H_3(X, \xi, t) &= 1.5751 \begin{bmatrix} \Delta a_{31}^3 & \Delta a_{32}^3 & \Delta a_{33}^3 \end{bmatrix}
\end{aligned} \tag{4.79}$$

$$\begin{aligned}
E_1(X, \xi, t) &= 1.4313 \Delta b_3^1 \\
E_2(X, \xi, t) &= 1.6004 \Delta b_3^2 \\
E_3(X, \xi, t) &= 1.5751 \Delta b_3^3 .
\end{aligned} \tag{4.80}$$

Based on the maximum bound of  $\Delta b_3^i(*)$  as given in Chapter 3, the norm of the function  $E_i(X, \xi, t)$  is obtained as follows :

$$\begin{aligned}
\|E_1(X, \xi, t)\| &= 0.0328 < 1 \\
\|E_2(X, \xi, t)\| &= 0.0249 < 1 \\
\|E_3(X, \xi, t)\| &= 2.6154 \times 10^{-4} < 1 .
\end{aligned} \tag{4.81}$$

Thus, equation (4.19) in assumption (ii) is satisfied.

To obtain the nonlinear part of the controller, the functions  $\rho_{L_i}(Z_i, \xi, t)$  in equation (4.34) were computed based on the maximum bound of the elements  $\Delta a_{ij}^i$  and  $\Delta b_i^i$ , and are given as the following :

$$\begin{aligned}
\rho_{L_1}(Z_1, \xi, t) &= 1.0339 \left\{ \|4.1174 X_2^1 + 0.5271 X_3^1\| + \|0.0328 w_1\| \right\} \\
\rho_{L_2}(Z_2, \xi, t) &= 1.0256 \left\{ \|28.1056 X_1^2 + 3.9371 X_2^2 + 0.1489 X_3^2\| \right. \\
&\quad \left. + \|0.0249 w_2\| \right\} \\
\rho_{L_3}(Z_3, \xi, t) &= 1.0003 \left\{ \|12.5947 X_1^3 + 0.3616 X_2^3 + 2.6029 \times 10^{-3} X_3^3\| \right. \\
&\quad \left. + \|2.6154 \times 10^{-4} w_3\| \right\} ,
\end{aligned} \tag{4.82}$$

where



$$\tilde{U}_i = K_i Z_i(t) + \Omega_i(t), \quad i = 1, 2, 3. \quad (4.83)$$

The following form of the nonlinear term (equation 4.32) in the decentralized control law has been used :

$$\Phi_{L_i}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_{L_i}(Z_i, \xi, t)}{\|\mu_{L_i}(Z_i, \xi, t)\|} \rho_{L_i}(Z_i, \xi, t) & \text{if } \|\mu_{L_i}(Z_i, \xi, t)\| > \epsilon_i \\ -\frac{\mu_{L_i}(Z_i, \xi, t)}{\epsilon_i} \rho_{L_i}(Z_i, \xi, t) & \text{if } \|\mu_{L_i}(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (4.84)$$

where  $\mu_{L_i}(Z_i, \xi, t)$  is as given by equation (4.33) for  $i=1, 2, 3$ .

The system was simulated to study the performance of the controller. Here, a fast trajectory was used. As in the previous section, the robot manipulator was required to move from an initial position of  $\theta(0) = [-0.8, -1.5, -0.5]^T$  radians to the final position of  $\theta(\tau) = [1.0, 0.2, 1.2]^T$  radians in  $\tau = 2$  seconds. The following values of  $\epsilon_i$  in equation (4.84) have been used :

$$\epsilon_1 = 1.1 \quad ; \quad \epsilon_2 = 0.5 \quad ; \quad \epsilon_3 = 0.9 \quad .$$

Figures 4.7, 4.8, and 4.9 illustrate the position, velocity and acceleration tracking responses of the manipulator, respectively, while carrying a 10 Kg. load. It can be seen that the robot manipulator tracks the desired trajectories (position, velocity and acceleration) for each joint with very small errors. Figure 4.10 shows the position tracking errors which verify the above observation. For comparison purposes, Figure 4.10 also shows the position tracking error for each joint of the robot manipulator using the linear independent joint controller as discussed in the previous section. It can be seen that the linear independent controller results in a larger tracking error as compared to the decentralized nonlinear controller. This is because the linear controller cannot compensate the uncertainties present in the system, as compared to the decentralized nonlinear controller.

### Joint Tracking Response For 10 Kg Load

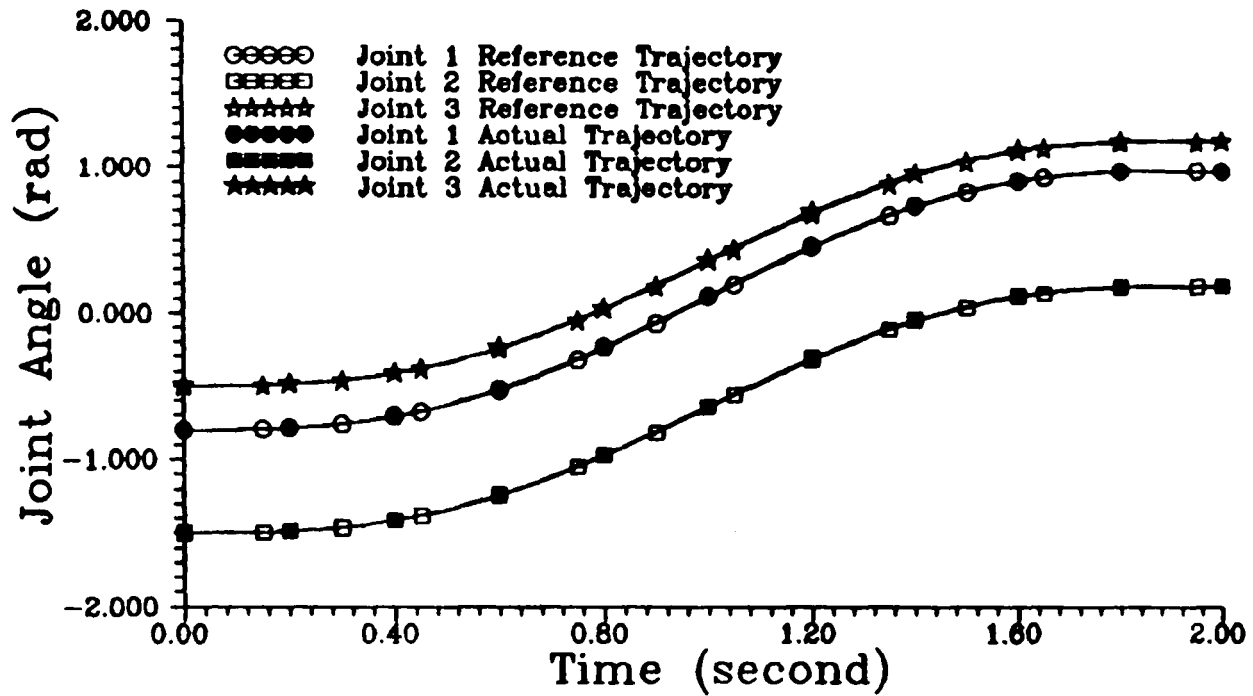


FIGURE 4.7 : Position Tracking Responses For 10 Kg. Load Using Decentralized Nonlinear Control Method

### Joint Tracking Response For 10 Kg Load

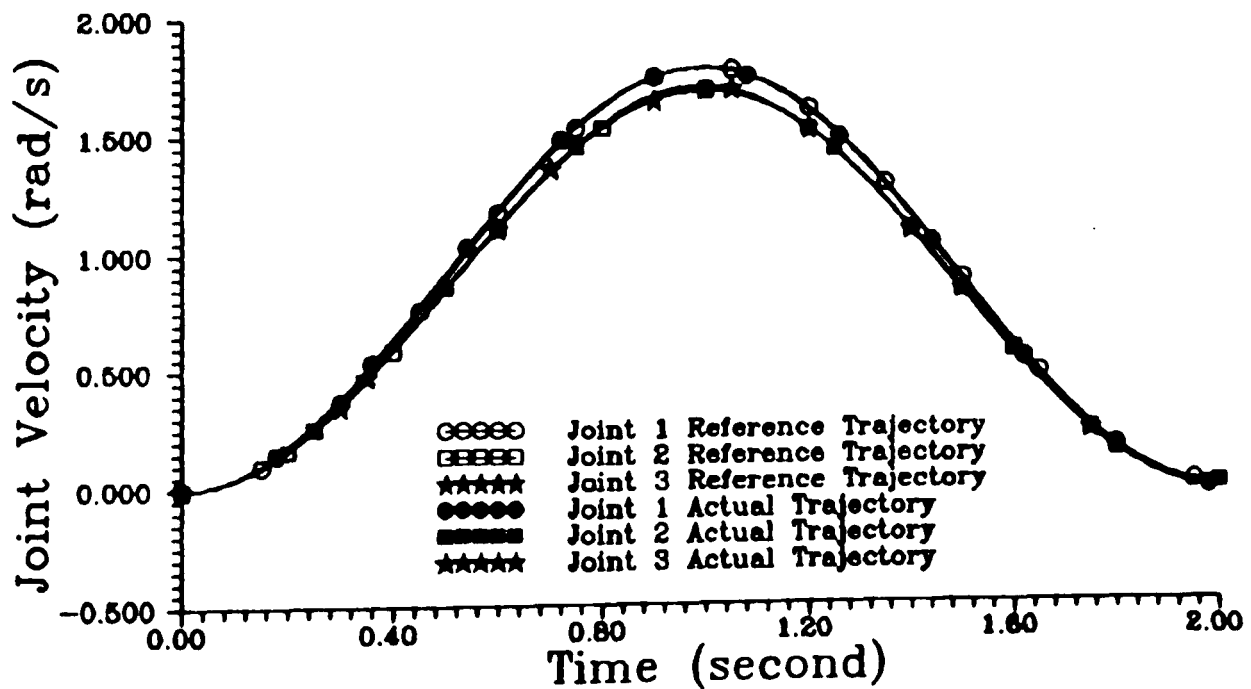


FIGURE 4.8 : Velocity Tracking Responses For 10 Kg. Load Using Decentralized Nonlinear Controller

### Joint Tracking Response For 10 Kg Load

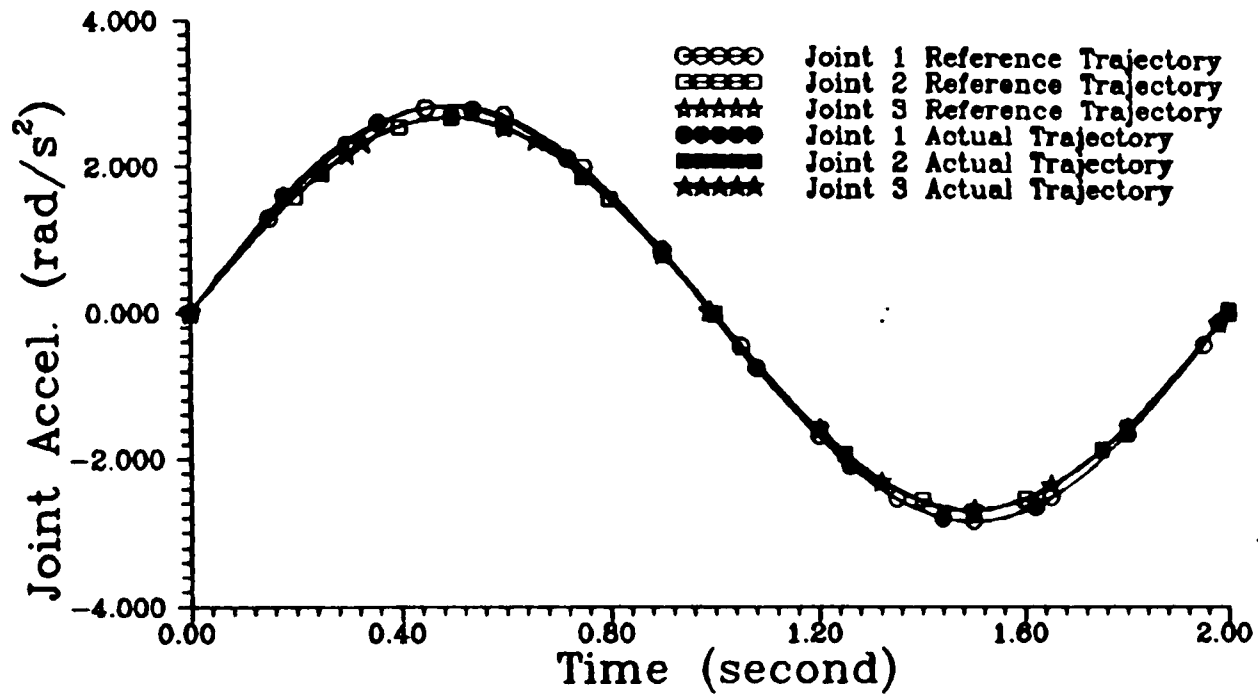


FIGURE 4.9 : Acceleration Tracking Responses For 10 Kg. Load Using Decentralized Nonlinear Control Method

### Joint Tracking Error For 10 Kg Load

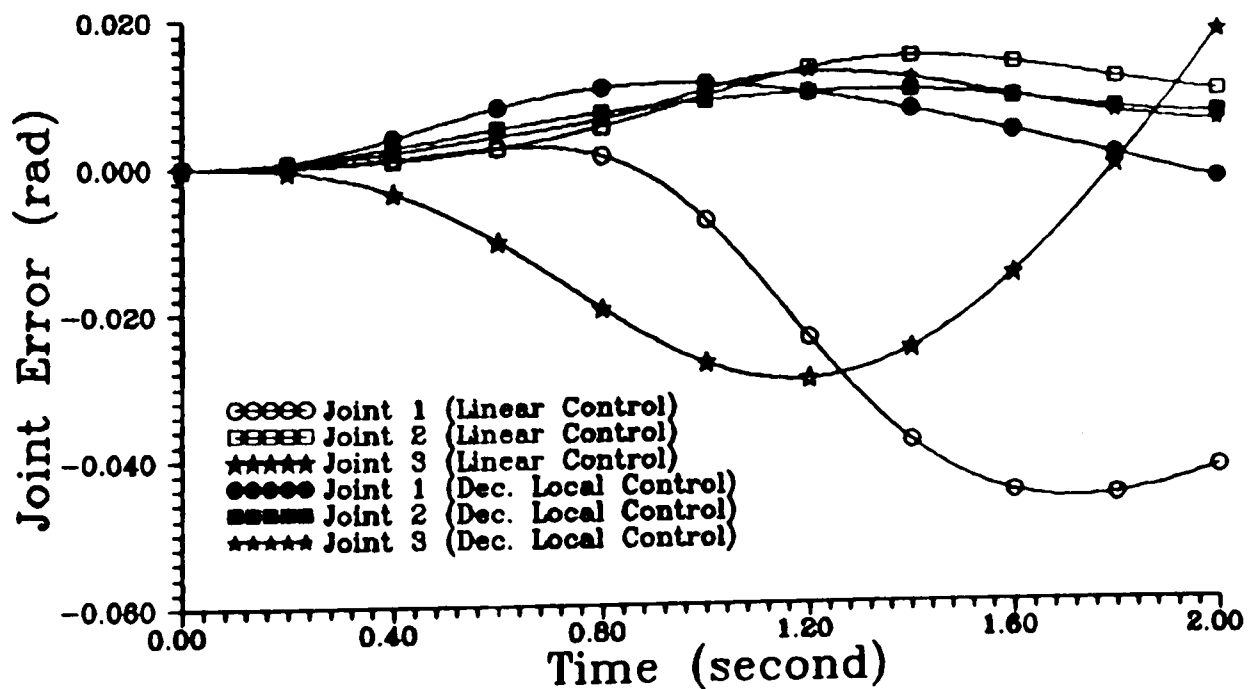
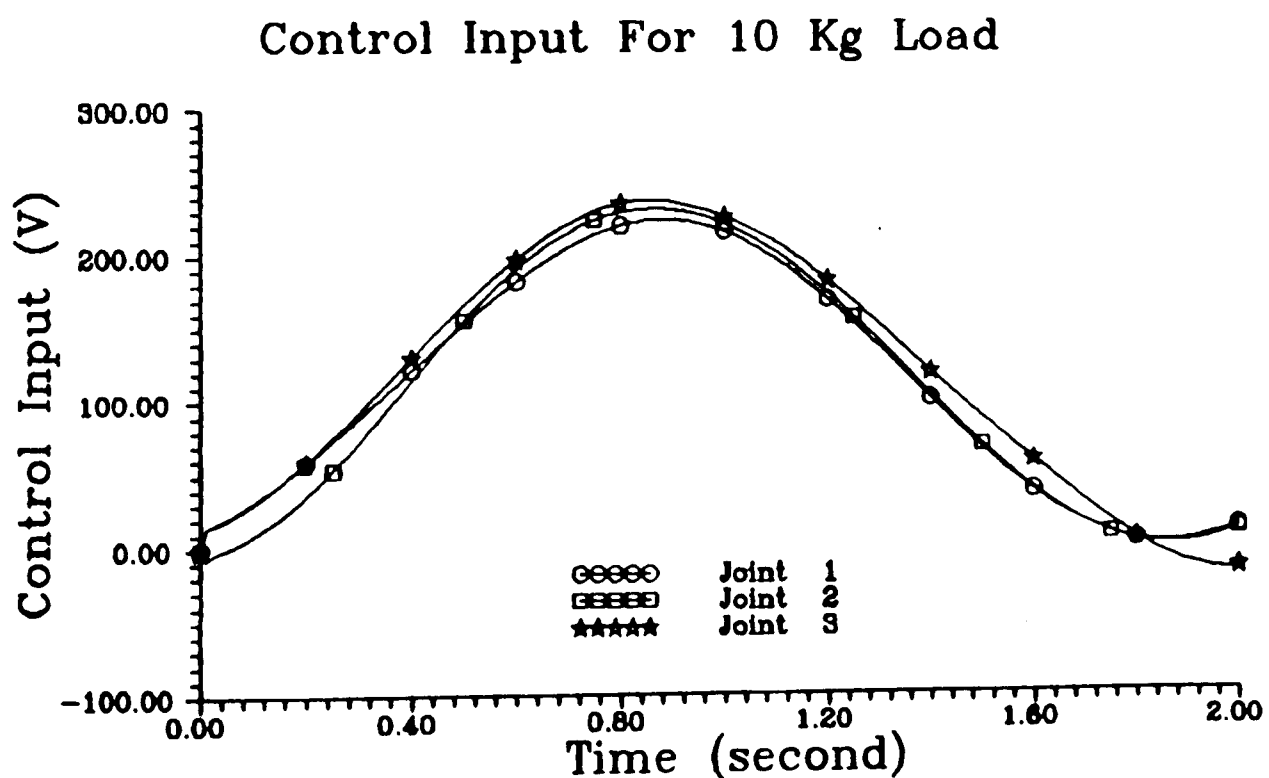


FIGURE 4.10 : Position Tracking Error For 10 Kg. Load

The simulation shows that under the same conditions, the performance of the decentralized nonlinear controller is better than the linear independent joint controller. The inclusion of the nonlinear component in the controller reduces the tracking error caused by the uncertainties and nonlinearities in the system.

Figure 4.11 illustrates the control input for each joint of the manipulator using the decentralized nonlinear control law. The control inputs are continuous and are within the operating limit of the actuators.

In the next set of simulations, the performance of the controller is evaluated under different loading conditions.



**FIGURE 4.11 : Joint Control Inputs For 10 Kg. Load Using Decentralized Nonlinear Control Method**

#### 4.5.3.1 Effect Of Load Variation

In the simulations, the manipulator was required to carry a 20 Kg. load, a 10 Kg. load and 0 Kg. load (no load), respectively. Figures 4.12, 4.13, and 4.14 illustrate the tracking performance of joint 1, joint 2, and joint 3 of the manipulator, respectively. Also shown in the figures are the joint tracking errors under the mentioned loading conditions.

It can be seen from the graphs that the controller is capable of forcing the robot manipulator to track the desired trajectories under different loading conditions with small tracking errors especially for joint 1. The tracking error for joints 2 and 3 are large compare to joint 1 for the 20 Kg. load in particular due to the gravitational forces acting on joints 2 and 3 (joint 1 does not have any gravitational forces acting on it). Nonetheless, the tracking error for joints 2 and 3 are small compared to the tracking error using the linear independent joint control method as shown in Figure 4.6 under the same conditions (20 Kg. load and fast trajectory).

The joint velocity and acceleration tracking responses under the different loading conditions are as shown in Figures 4.15, 4.16, and 4.17 for joints 1, 2, and 3, respectively. The figures show that the joint velocity and acceleration tracks the desired joint velocity and acceleration profile with a small tracking error especially for joints 1 and 3. The simulations show that the decentralized nonlinear control law is robust to the parameter variations as a result of the different loading conditions. It should be noted that the decentralized nonlinear controller was designed to operate for the payload range of 0Kg. to 20 Kg. Thus, for any load carried by the manipulator that is within this range, the robot manipulator is capable of tracking a given desired trajectory satisfactorily.

Figure 4.18 and Figure 4.19 illustrate the control inputs to the joint actuators when the manipulator is moving without carrying any load and while carrying a 20 Kg. load, respectively. The control inputs produced by the decentralized nonlinear controller are continuous and smooth for both cases. There is very little difference between the two cases. The control inputs are within the specified limit of the actuators, and are at the maximum when the motion of the manipulator is at its maximum velocity as expected.

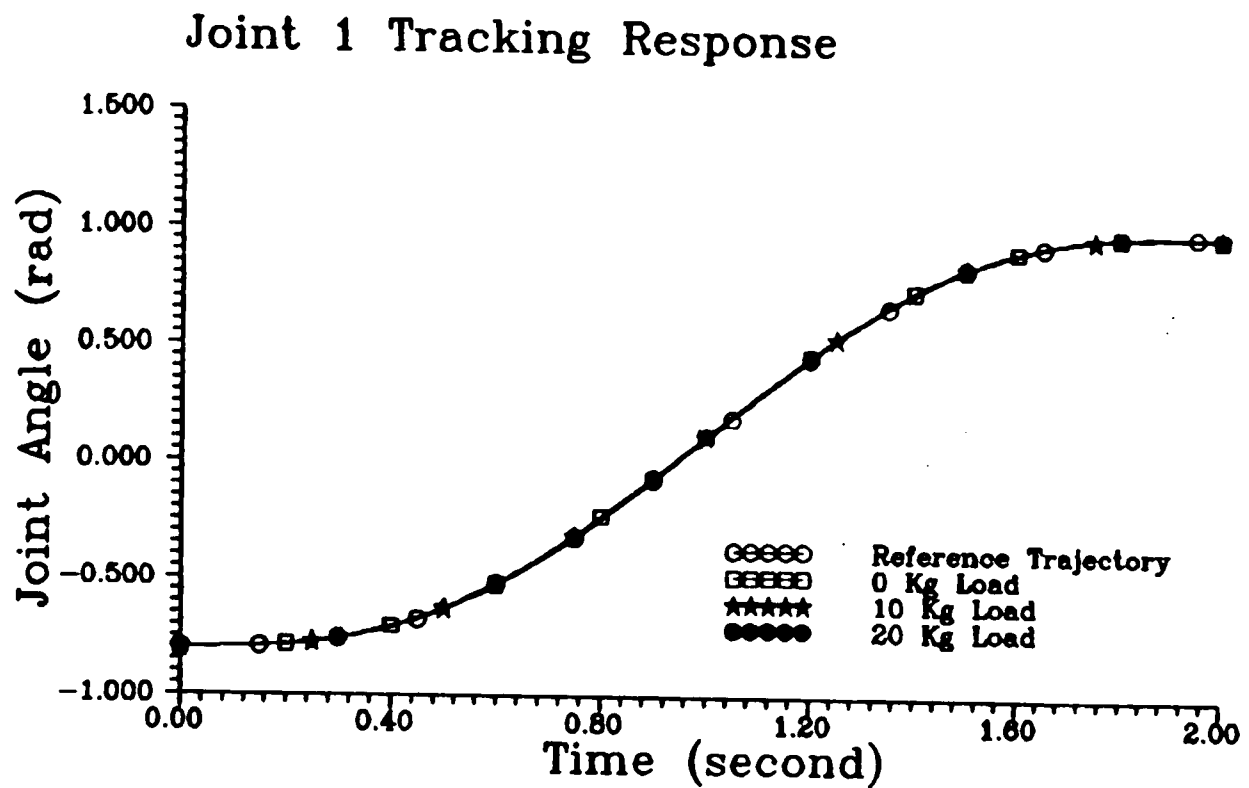


FIGURE 4.12 a : Joint 1 Tracking Response Under Different Load  
Using Decentralized Nonlinear Control Method

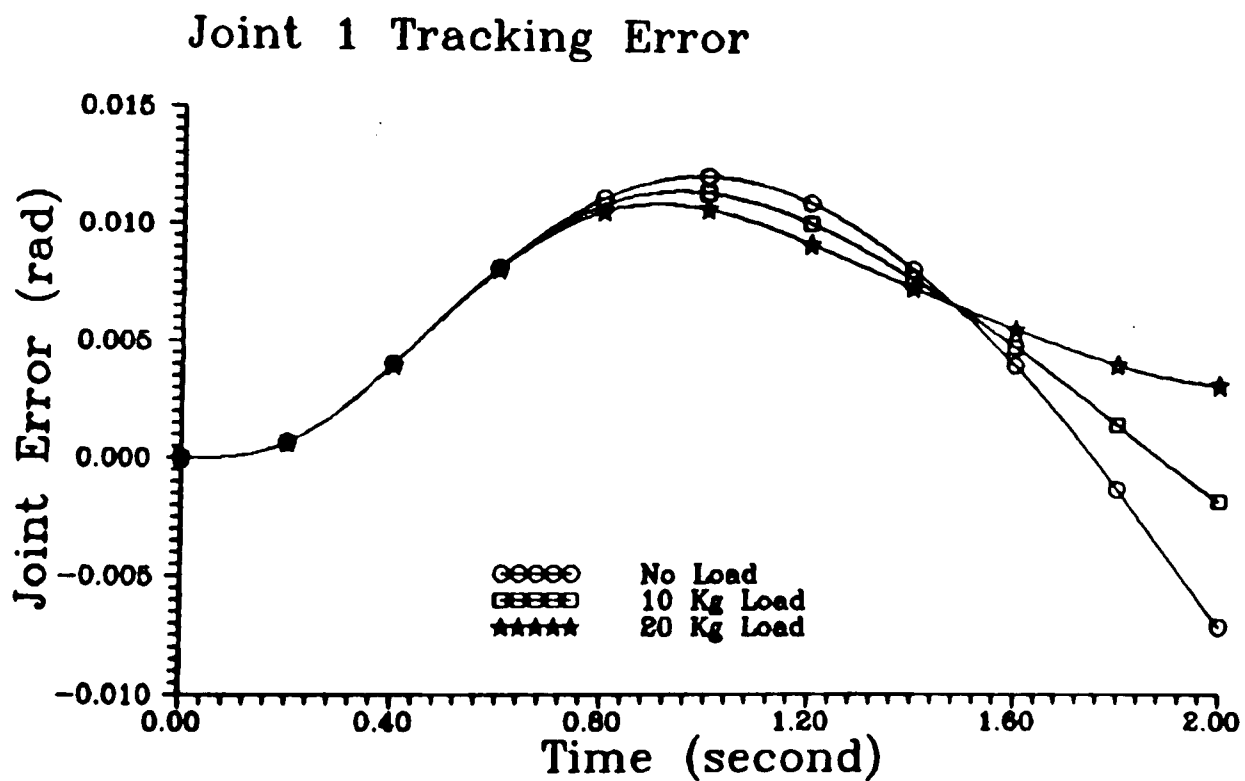


FIGURE 4.12 b : Joint 1 Tracking Error Under Different Load  
Using Decentralized Nonlinear Control Method

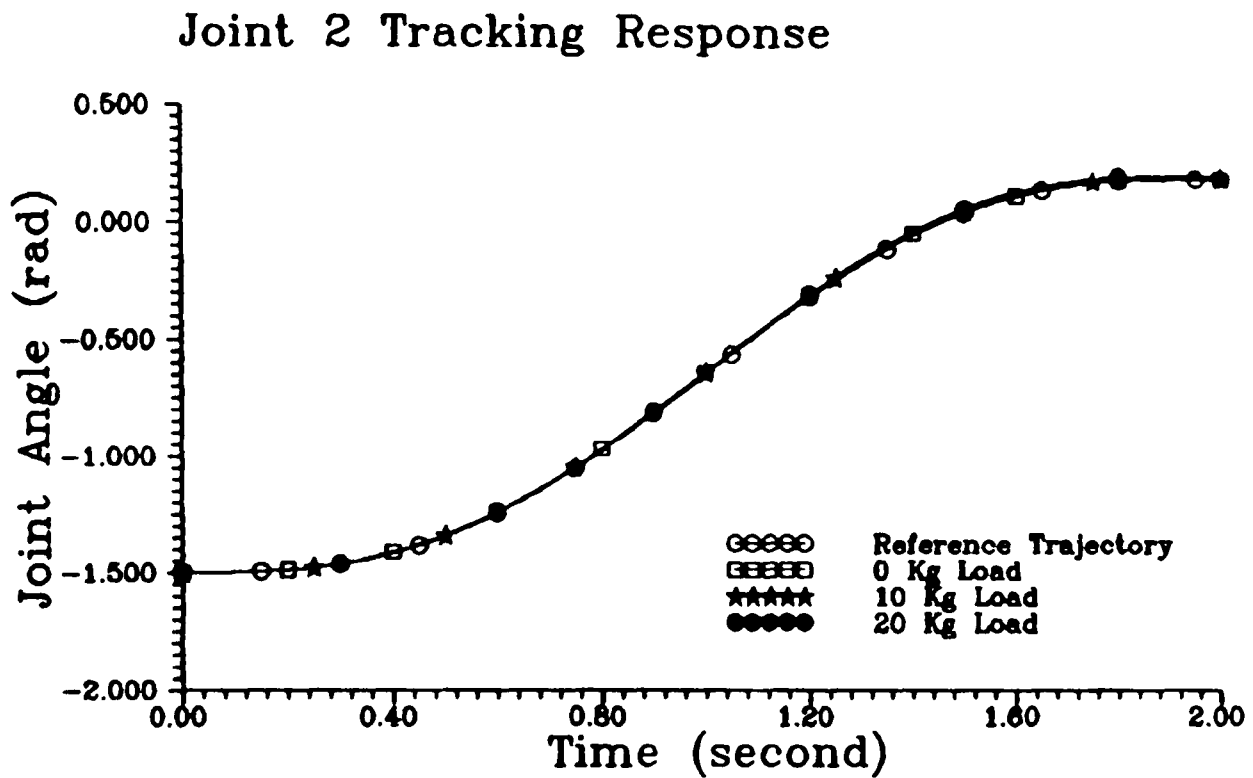


FIGURE 4.13 a : Joint 2 Tracking Response Under Different Load  
Using Decentralized Nonlinear Control Method

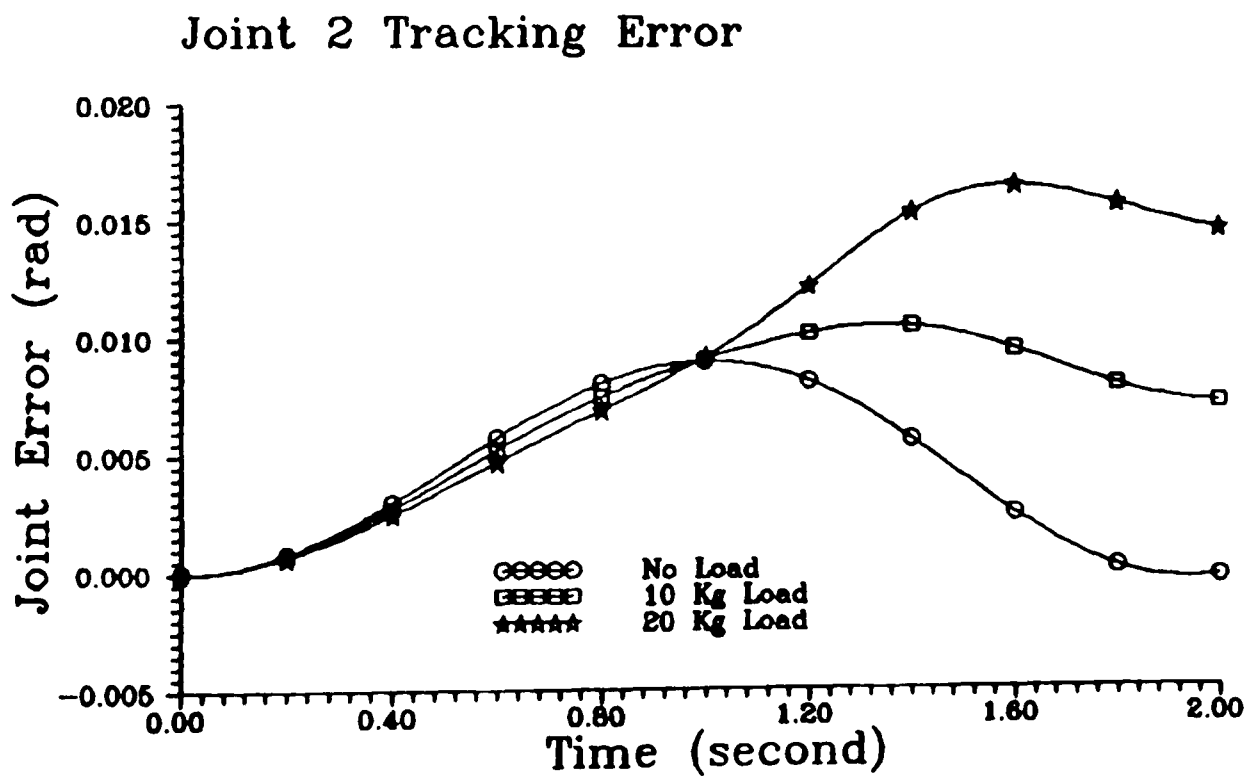


FIGURE 4.13 b : Joint 2 Tracking Error Under Different Load  
Using Decentralized Nonlinear Control Method

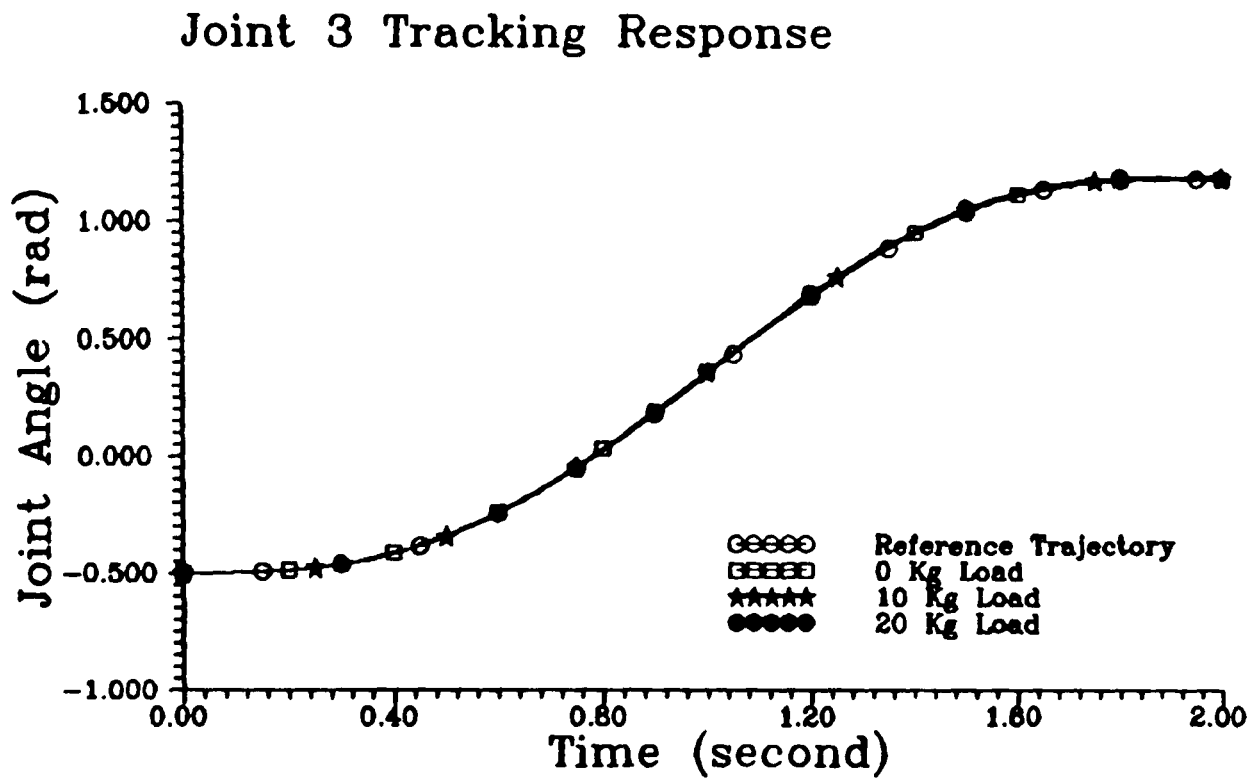


FIGURE 4.14 a : Joint 3 Tracking Response Under Different Load  
Using Decentralized Nonlinear Control Method

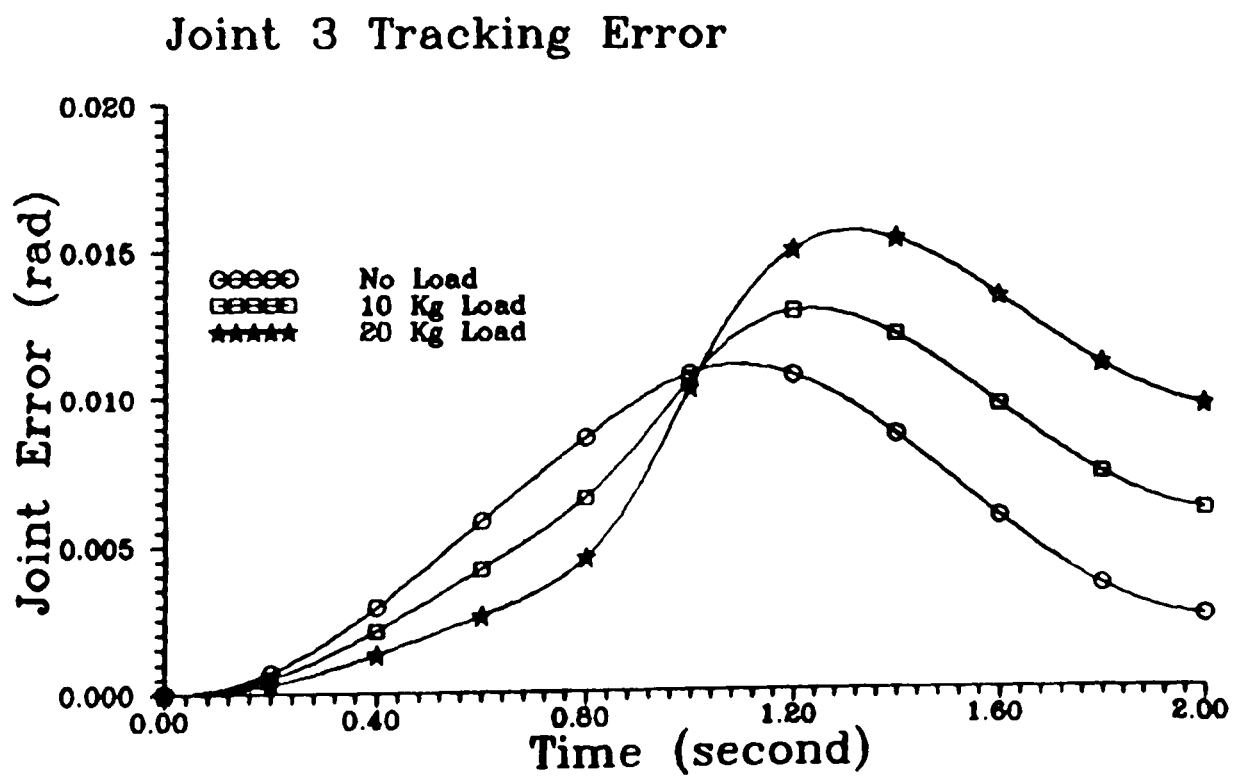


FIGURE 4.14 b : Joint 3 Tracking Error Under Different Load  
Using Decentralized Nonlinear Control Method



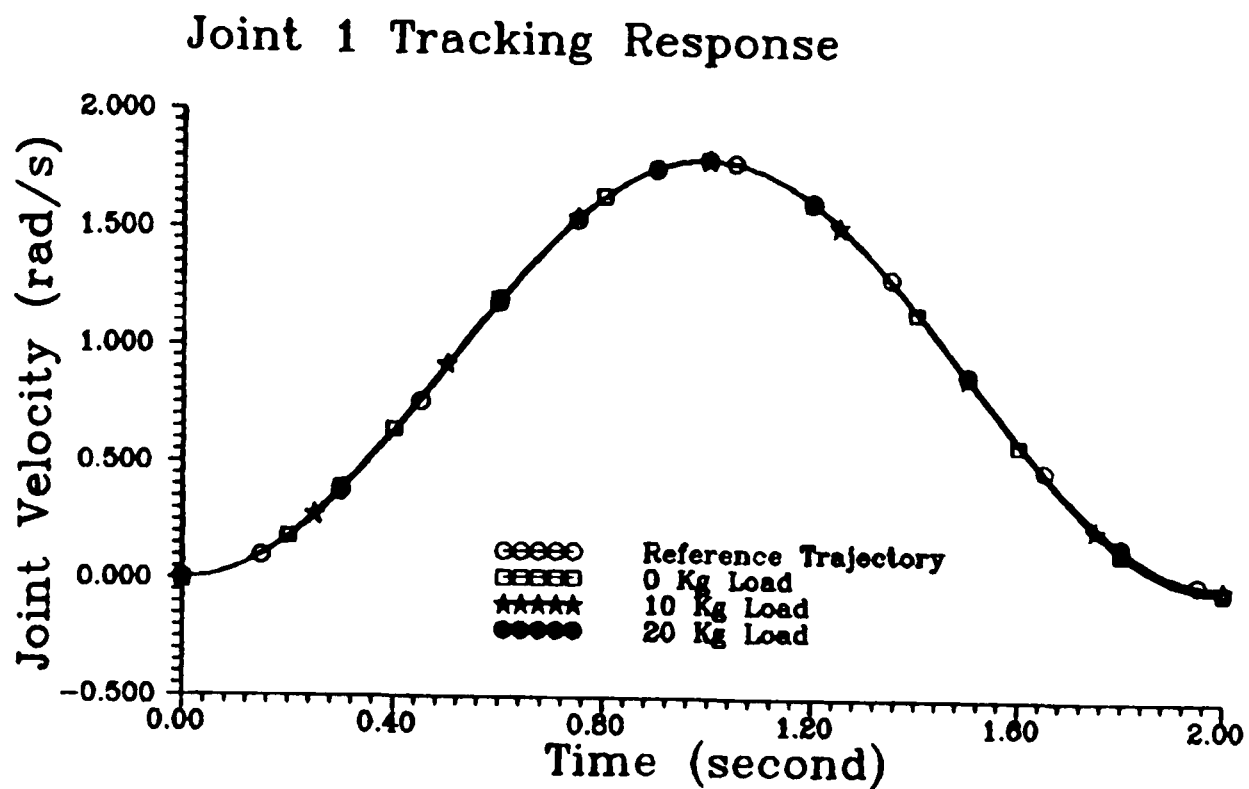


FIGURE 4.15 a : Joint 1 Velocity Response Under Different Load  
Using Decentralized Nonlinear Control Method

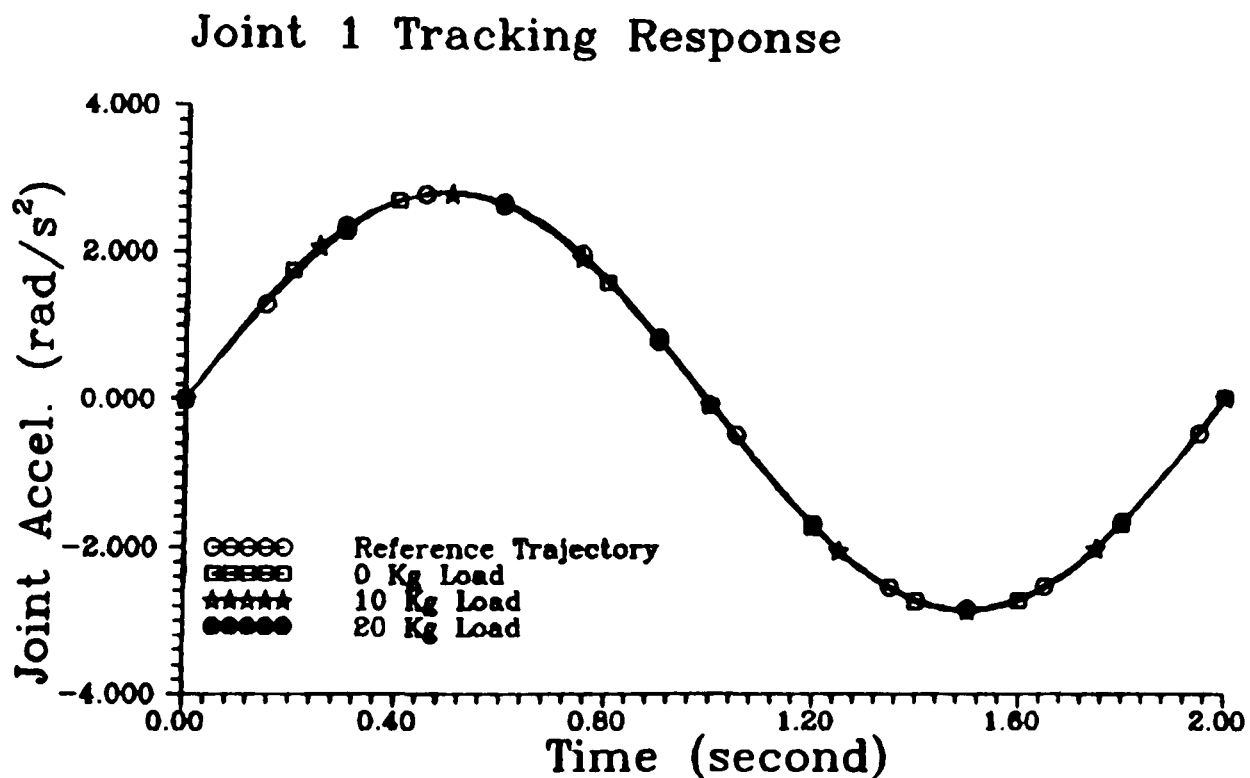


FIGURE 4.15 b : Joint 1 Acceleration Response Under Different Load  
Using Decentralized Nonlinear Control Method

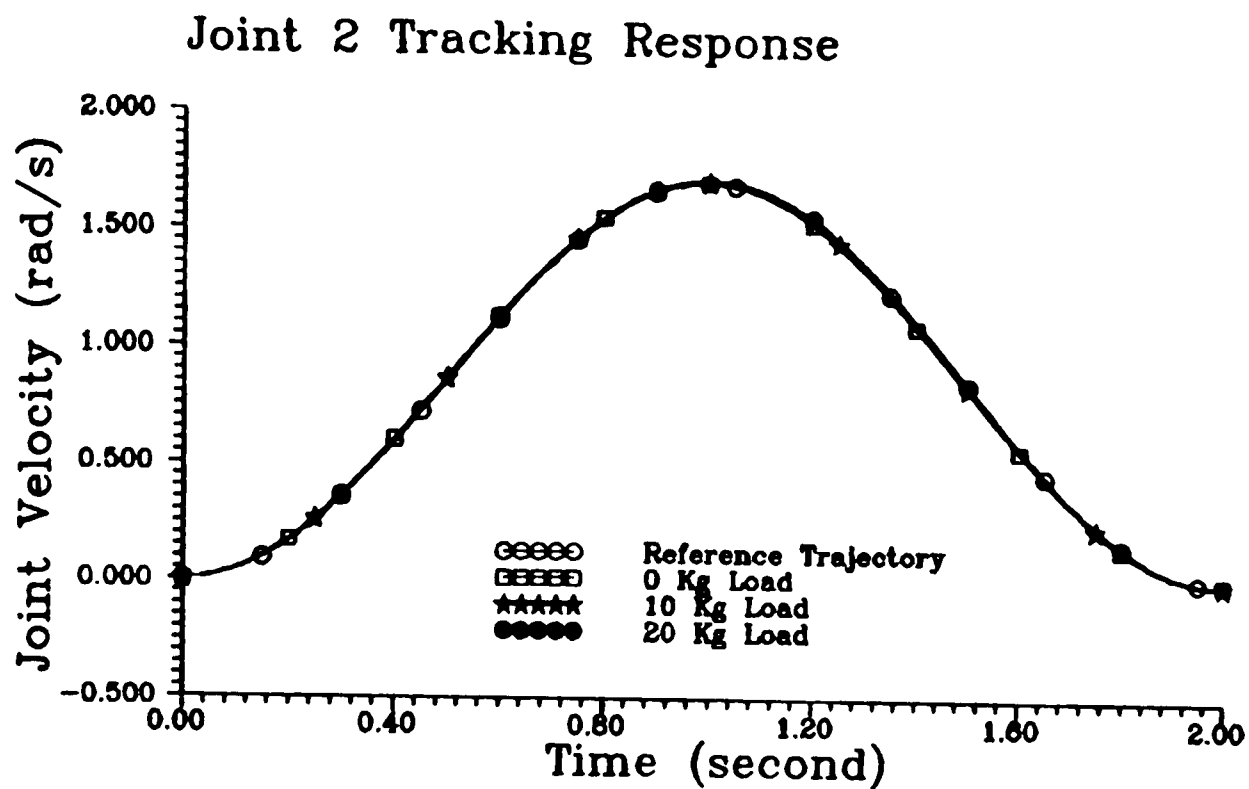


FIGURE 4.16 a : Joint 2 Velocity Response Under Different Load Using Decentralized Nonlinear Control Method

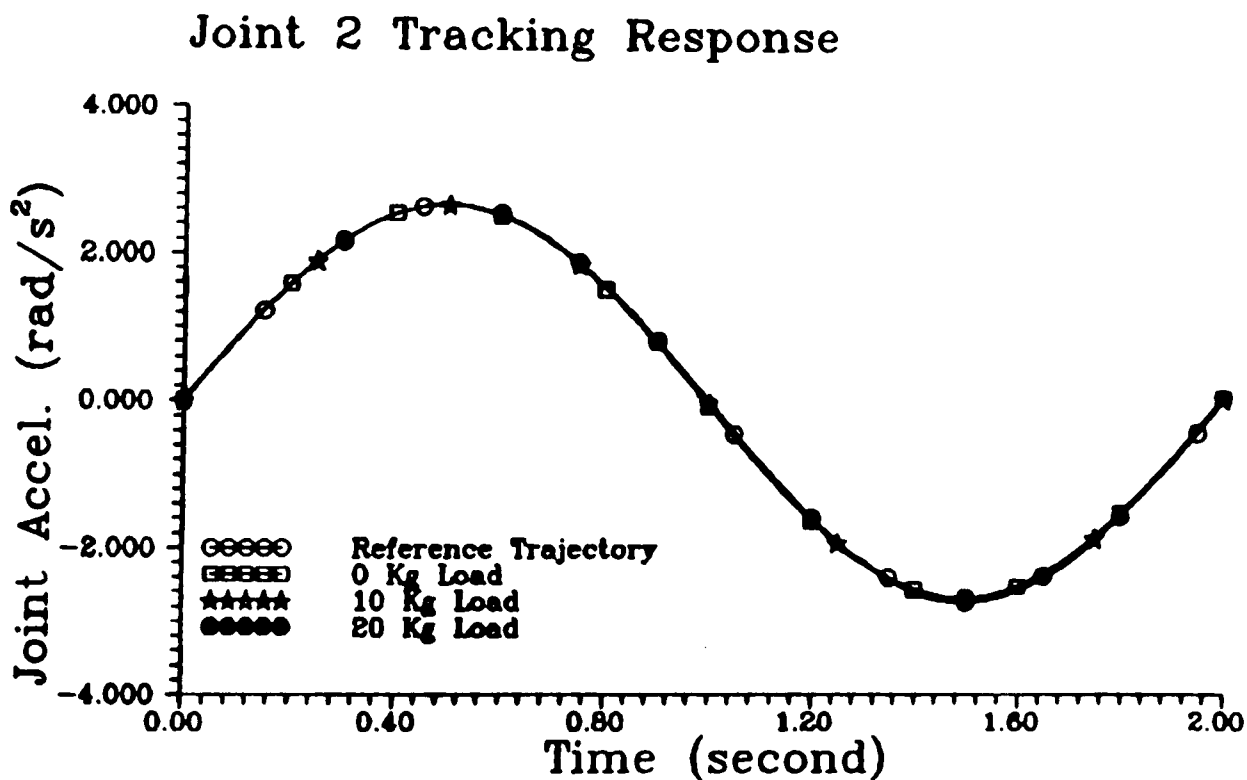


FIGURE 4.16 b : Joint 2 Acceleration Response Under Different Load Using Decentralized Nonlinear Control Method

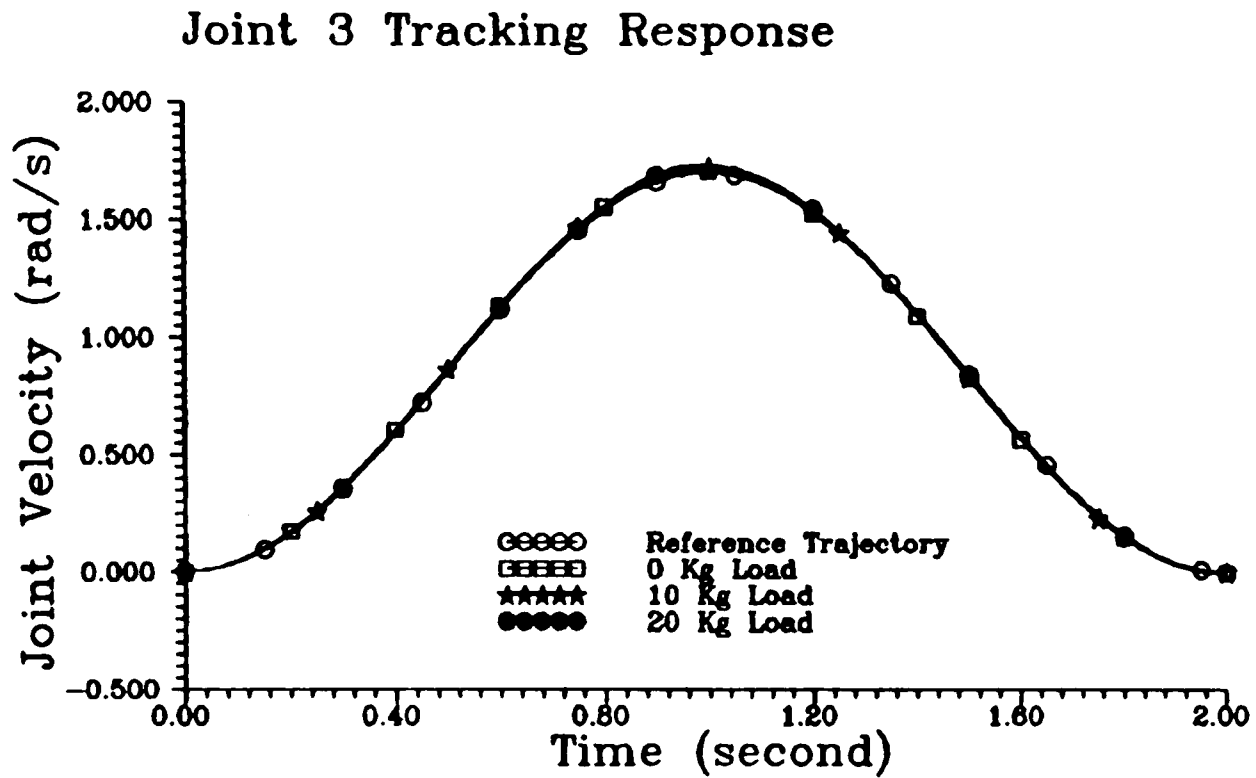


FIGURE 4.17 a : Joint 3 Velocity Response Under Different Load Using Decentralized Nonlinear Control Method

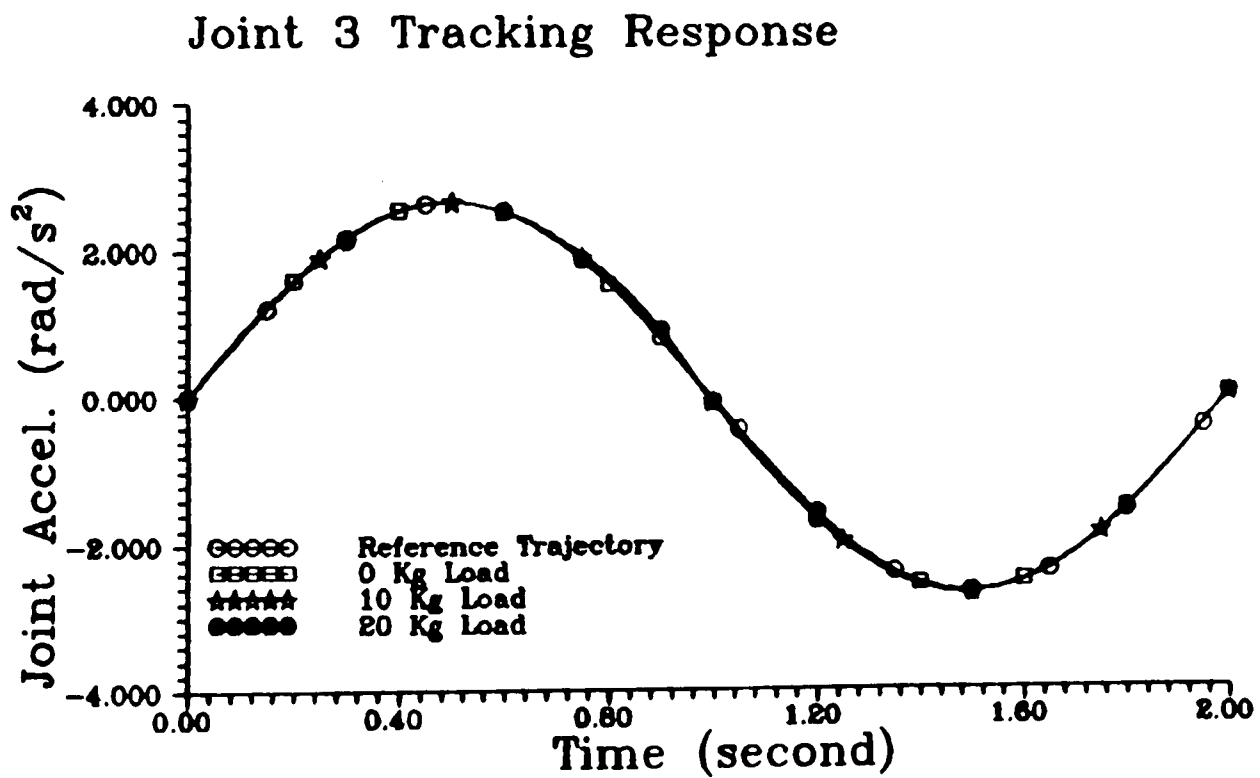


FIGURE 4.17 b : Joint 3 Acceleration Response Under Different Load Using Decentralized Nonlinear Control Method

Control Input For 0 Kg Load

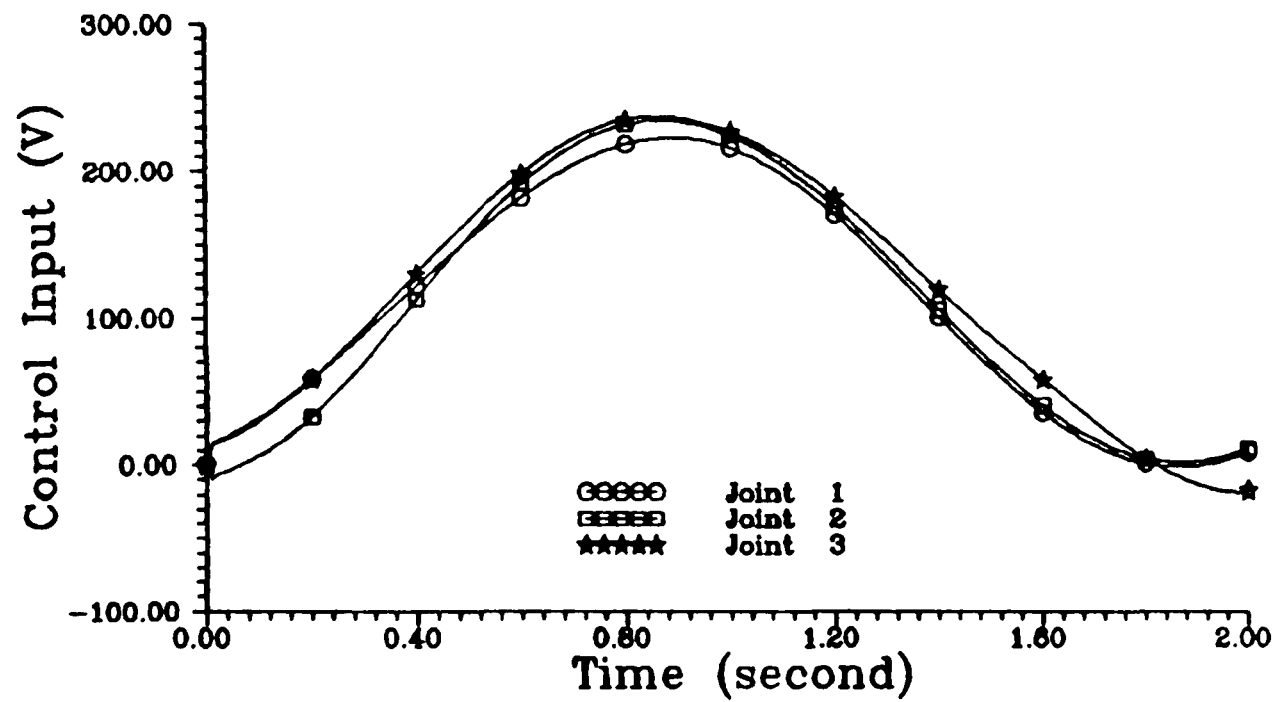


FIGURE 4.18 : Control Inputs For 0 Kg. Load Using Decentralized Nonlinear Control Method

Control Input For 20 Kg Load

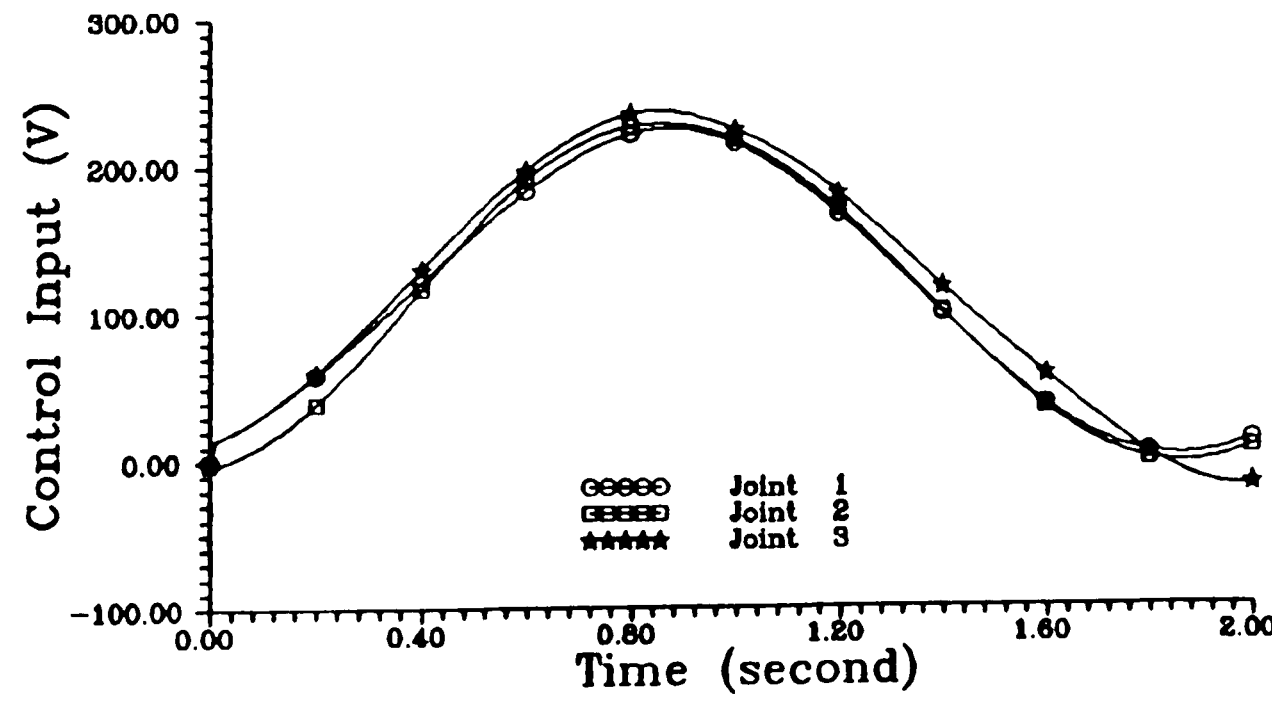


FIGURE 4.19 : Control Inputs For 20 Kg. Load Using Decentralized Nonlinear Control Method

In the next subsection, the effect of varying the constant parameter  $\epsilon_i$  in equation (4.32) is studied.

#### 4.5.3.2 Effect Of Varying The Value $\epsilon_i$

In the following simulations, the effect of varying the constant  $\epsilon_i$  in the nonlinear control component is studied.  $\epsilon_i$  can be viewed as a thin boundary layer at which the nonlinear control  $\Phi_{Li}(Z_i, \xi, t)$  switches its form.

It was desired for the manipulator carrying a 10 Kg. load to track a trajectory starting from an initial position of  $\theta(0) = [ -0.5 , -1.2 , -0.2 ]^T$  radians to the final position of  $\theta(\tau) = [ 1.0 , 0.2 , 1.2 ]^T$  radians in  $\tau = 2$  seconds.

The following values of  $\epsilon_i$  have been considered in the simulations :

Case 1 :  $\epsilon_1 = 2.0$  ,  $\epsilon_2 = 2.0$  ,  $\epsilon_3 = 2.0$  ;

Case 2 :  $\epsilon_1 = 1.5$  ,  $\epsilon_2 = 1.5$  ,  $\epsilon_3 = 1.5$  ;

Case 3 :  $\epsilon_1 = 1.0$  ,  $\epsilon_2 = 1.0$  ,  $\epsilon_3 = 1.0$  ;

Case 4 :  $\epsilon_1 = 0.25$  ,  $\epsilon_2 = 0.2$  ,  $\epsilon_3 = 0.54$  .

Figures 4.20, 4.21, and 4.22 illustrate the tracking error for joint 1, joint 2 and joint 3, respectively, for all the above cases. It can be seen from the figures that, as the values of the  $\epsilon_i$  decreases, the tracking error decreases for all the joints. Hence, the tracking accuracy of the robot manipulator increases as the  $\epsilon_i$  decreases. In other words, the smaller the value of  $\epsilon_i$ , the tighter the tracking, and the smaller the ultimate boundedness set given by equation (4.63). This is due to the fact that the radius  $\eta_L$  of the smallest closed ball given by equation (4.62) enclosed by the ultimate boundedness set decreases as the value of  $\epsilon_i$  decreases.

The control input for each joint for the Case 4 is as shown in Figure 4.23. Notice that there is a very small oscillation in the control input for joint 2 and

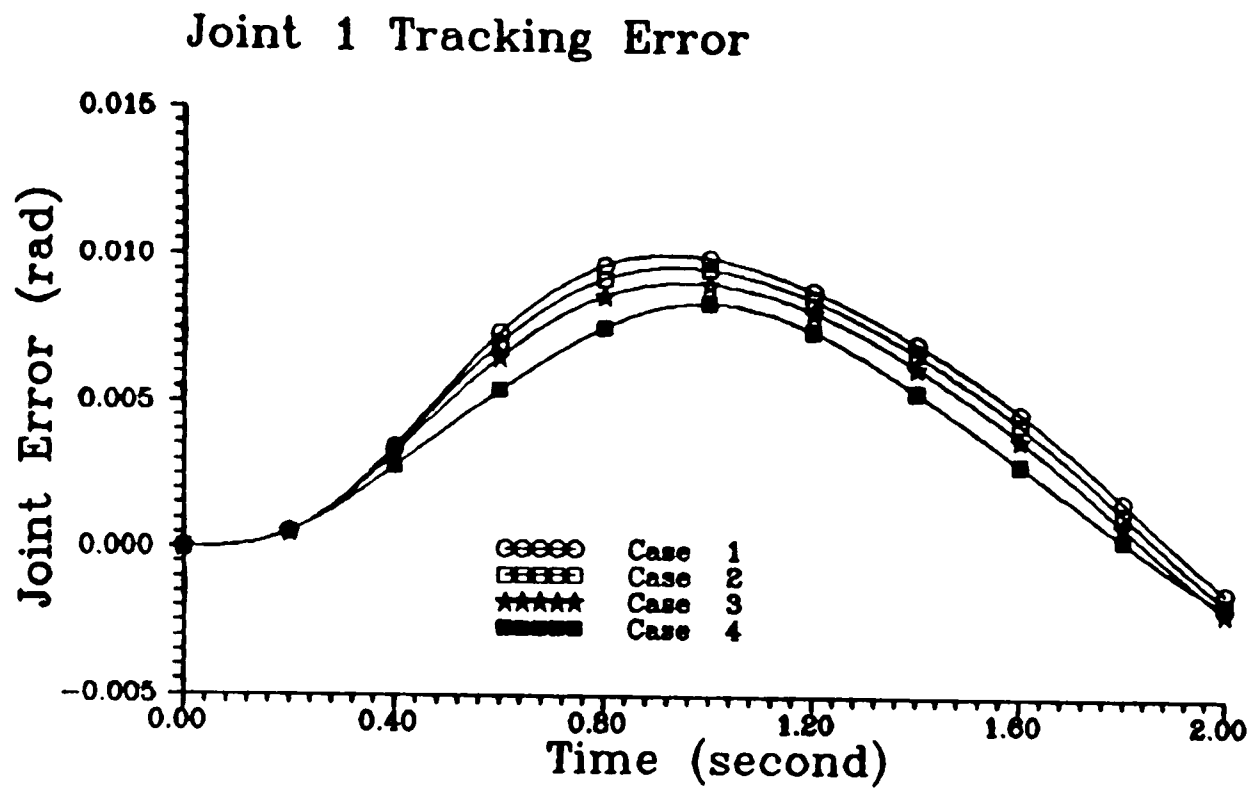


FIGURE 4.20 : Joint 1 Tracking Error For 10 Kg. Load  
For Different Values of  $\epsilon_i$

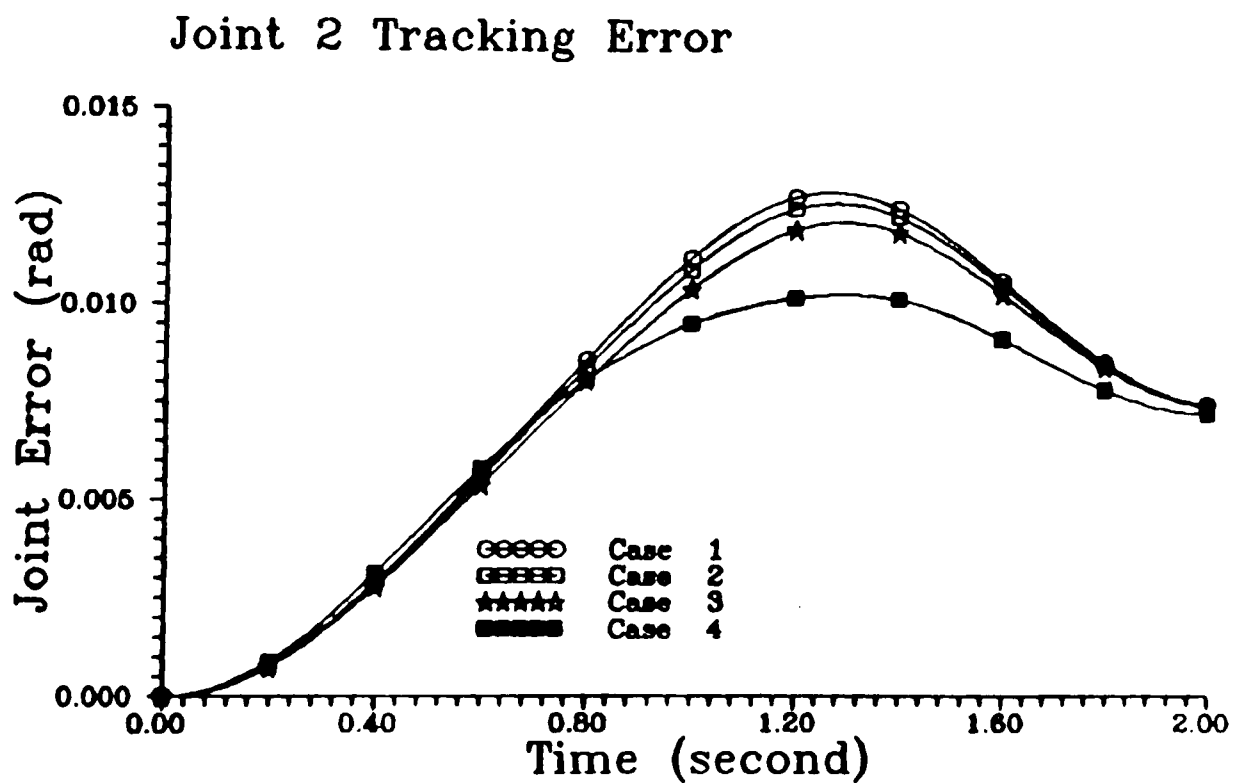


FIGURE 4.21 : Joint 2 Tracking Error For 10 Kg. Load  
For Different Values Of  $\epsilon_i$

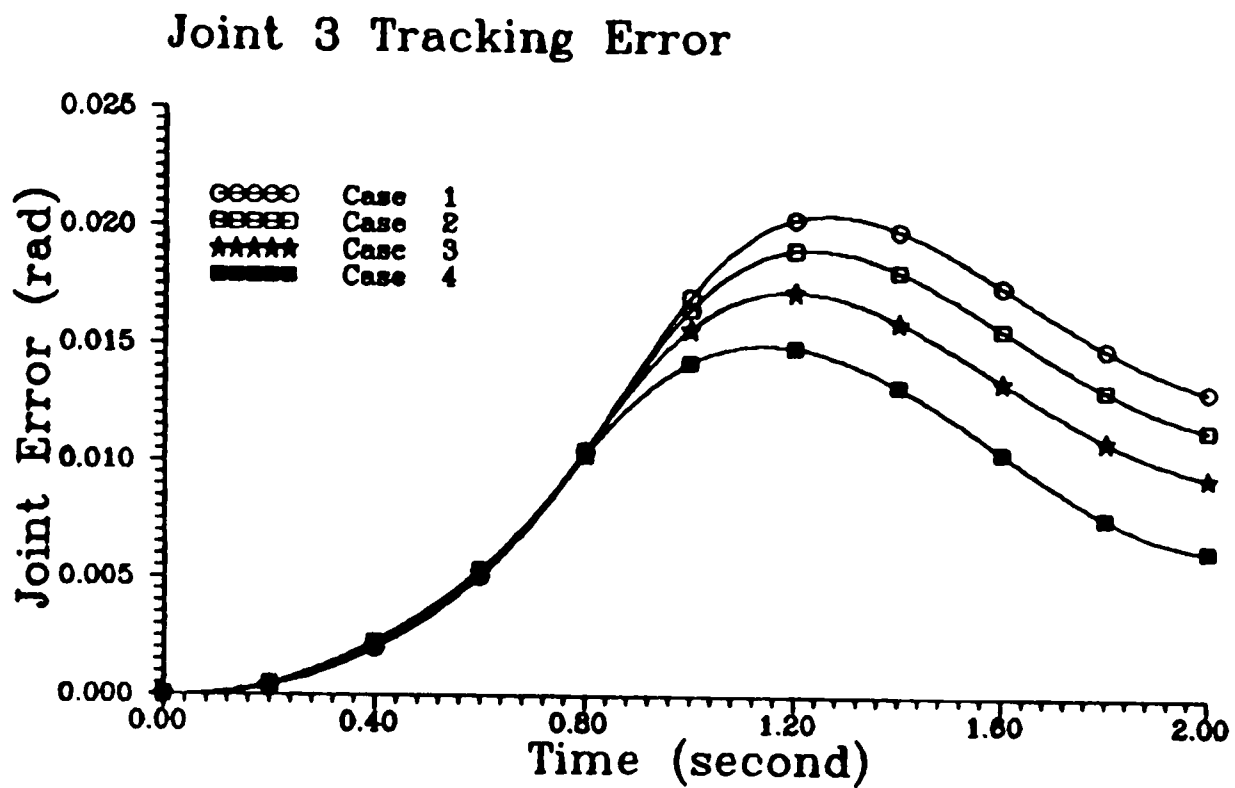


FIGURE 4.22 : Joint 3 Tracking Error For 10 Kg. Load  
For Different Values of  $\epsilon_i$

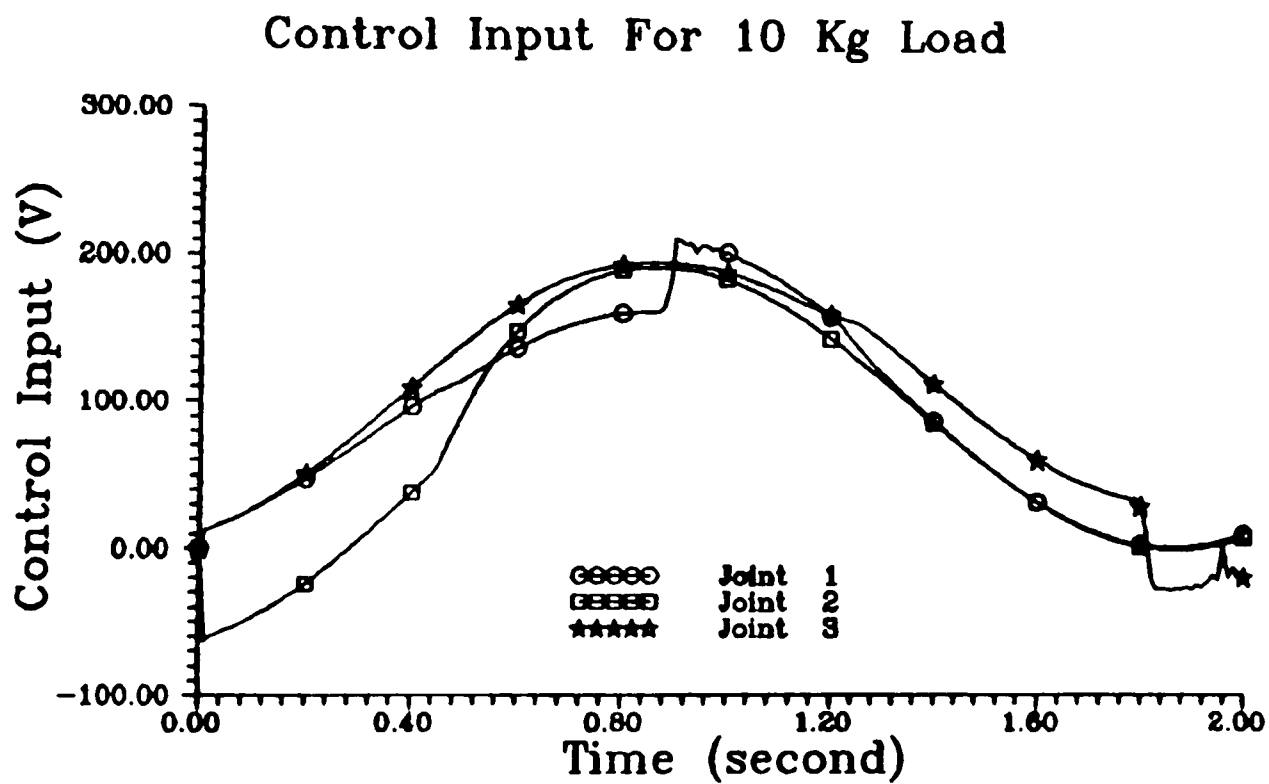


FIGURE 4.23 : Joint Control Inputs For 10 Kg. Load  
For  $\epsilon_1=0.25$  ,  $\epsilon_2=0.2$  ,  $\epsilon_3=0.54$

joint 3. The oscillation will become apparent and significant when the value of  $\epsilon_i$  is decreased further. This is illustrated by Figures 4.24 and 4.25. For these figures, the simulation was carried out with the values of  $\epsilon_i$  chosen as follows :

$$\epsilon_1 = 0.2 \quad , \quad \epsilon_2 = 0.15 \quad , \quad \epsilon_3 = 0.5 \quad .$$

Figure 4.24 demonstrates the tracking error of the robot manipulator with the specified value  $\epsilon_i$ , while Figure 4.25 illustrates the corresponding control input for each joint of the robot manipulator. As shown in Figure 4.25, the control inputs exhibit a phenomenon called chattering, which arises due to the digital implementation of the control law. The chattering occurs because the function  $\mu_{Li}(Z_i, \xi, t)$  changes in sign at the sampling frequency if the sampling interval is not small enough for a fixed value of  $\epsilon_i$ . Chattering is undesirable because it may excite the unmodelled dynamic effects of the robot manipulator such as the joint flexibility. It may also degrades the tracking performance of the robot manipulator as can be seen by comparing Figure 4.24 with Figures 4.20 to 4.22.

Thus, there is a trade-off between tracking accuracy and chattering of the control signal. Therefore, in the selection of an appropriate  $\epsilon_i$ , careful attention must be given to the desired tracking accuracy and the chattering of the control signals for a fixed sampling period.

In the following subsection, the effect of varying the weighting matrix  $Q_i$  in equation (4.36) is studied.

#### 4.5.3.3 Effect Of Varying The Weighting Matrix $Q_i$

In this set of simulations, the value of the weighting matrix  $Q_i$  in the matrix Lyapunov equation (4.36) was varied to evaluate the performance of the controller. The same feedback gains as in section 4.5.3.1 were used. Similar to the previous section, it was desired for the manipulator carrying a 10 Kg. load to track a trajectory starting from an initial position of  $\theta(0) = [ -0.5 \quad , \quad -1.2 \quad , \quad -0.2 ]^T$  radians to the final position of  $\theta(\tau) = [ 1.0 \quad , \quad 0.2 \quad , \quad 1.2 ]^T$  radians in  $\tau = 2$  seconds. The following value of  $\epsilon_i$  has been used :



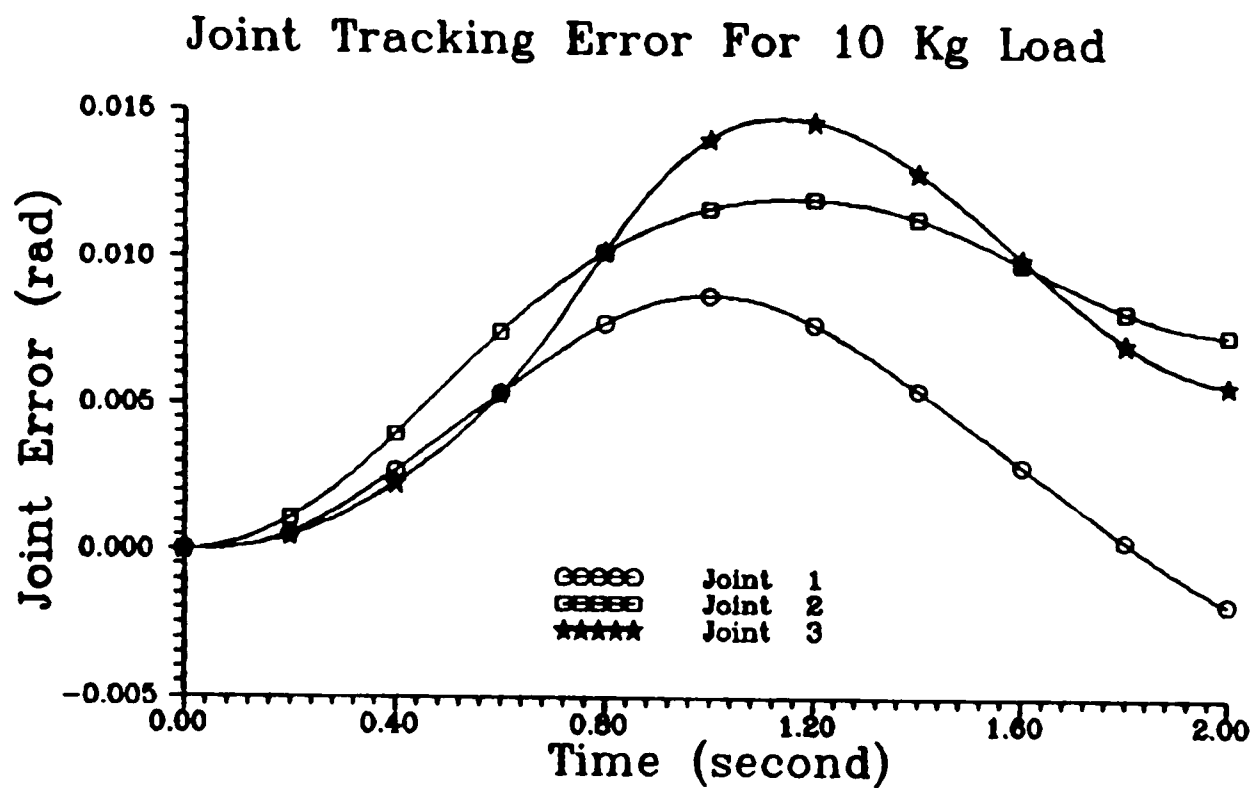


FIGURE 4.24 : Joint Tracking Errors For 10 Kg. Load  
For  $\epsilon_1=0.2$  ,  $\epsilon_2=0.15$  ,  $\epsilon_3=0.5$

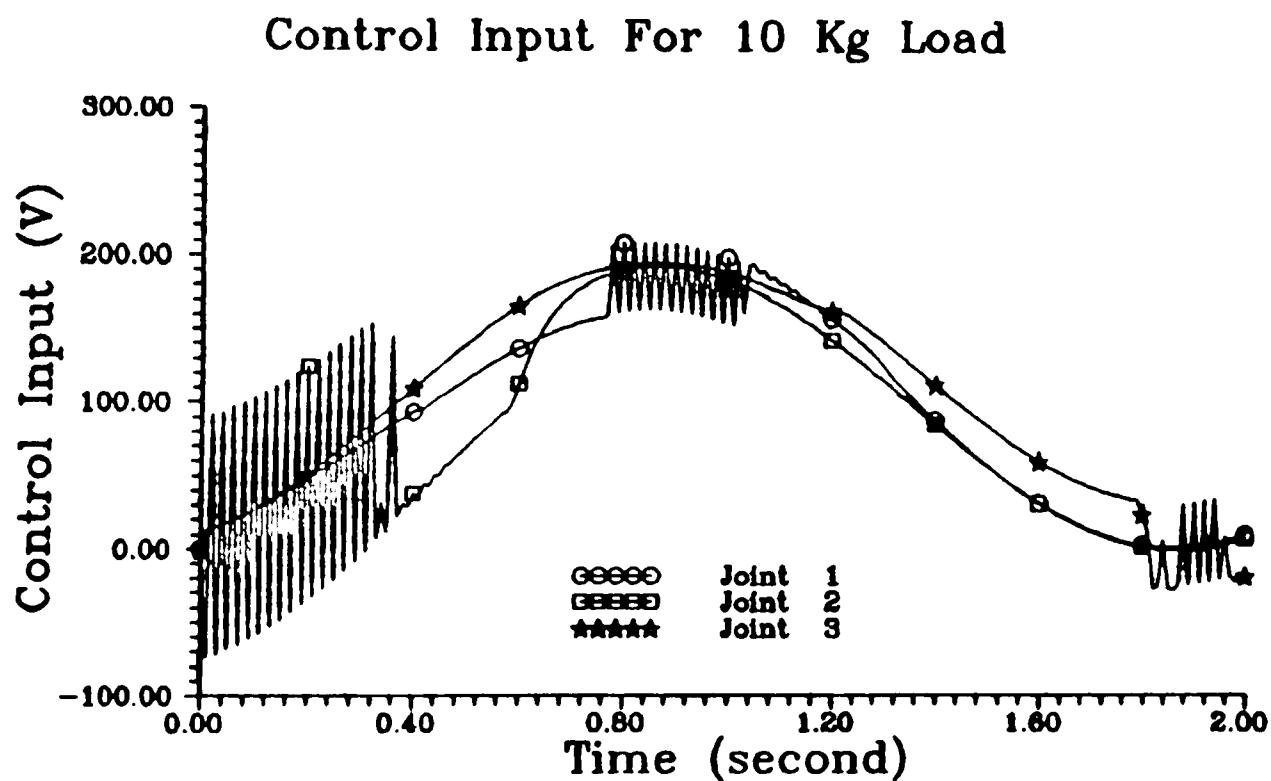


FIGURE 4.25 : Joint Control Inputs For 10 Kg. Load  
For  $\epsilon_1=0.2$  ,  $\epsilon_2=0.15$  ,  $\epsilon_3=0.5$

$$\epsilon_1 = 0.25 \quad , \quad \epsilon_2 = 0.2 \quad , \quad \epsilon_3 = 0.54 \quad .$$

For simplicity, it was assumed that the matrix  $Q_i$  was the same for all the three subsystems. The following values of  $Q_i$  have been considered :

$$\text{Case 1} \quad : \quad Q_i = 0.1 I_3 \quad ;$$

$$\text{Case 2} \quad : \quad Q_i = 0.5 I_3 \quad ;$$

$$\text{Case 3} \quad : \quad Q_i = I_3 \quad ;$$

$$\text{Case 4} \quad : \quad Q_i = 1.5 I_3 \quad ,$$

where  $I_3$  is a 3x3 identity matrix, and  $i = 1, 2, 3$ .

Figures 4.26, 4.27, and 4.28 demonstrate the joint tracking errors for joint 1, joint 2, and joint 3, respectively, for all the cases considered. It can be observed from the Figures that the tracking error for each joint decreases as the matrix  $Q_i$  increases in value. Thus, the tracking accuracy of the robot manipulator is proportional to the matrix  $Q_i$ . The simulation results agree with the theory (equation 4.63) in that the radius of the ultimate boundedness set decreases as  $Q_i$  increases.

Figure 4.29 illustrates the control input for each joint of the robot manipulator for Case 1, that is, when  $Q_i = 1.1 I_3$  for all the subsystems. It can be seen that chattering occur in all of the control signals. Thus, care should be taken in selecting an appropriate  $Q_i$  matrix for all the subsystems to ensure good tracking accuracy and at the same time to obtain smooth control signals.

The simulation results demonstrate that the proposed controller can accurately control the movement of the robotic arm with good robustness properties, and forces the robot manipulator to track the reference trajectory precisely in spite of the substantial uncertainties and nonlinearities that exist in the system. In general, the addition of the nonlinear control component improves the performance of the system as well as stabilizes a response that may be unstable in the presence of only the linear control component. It should be noted

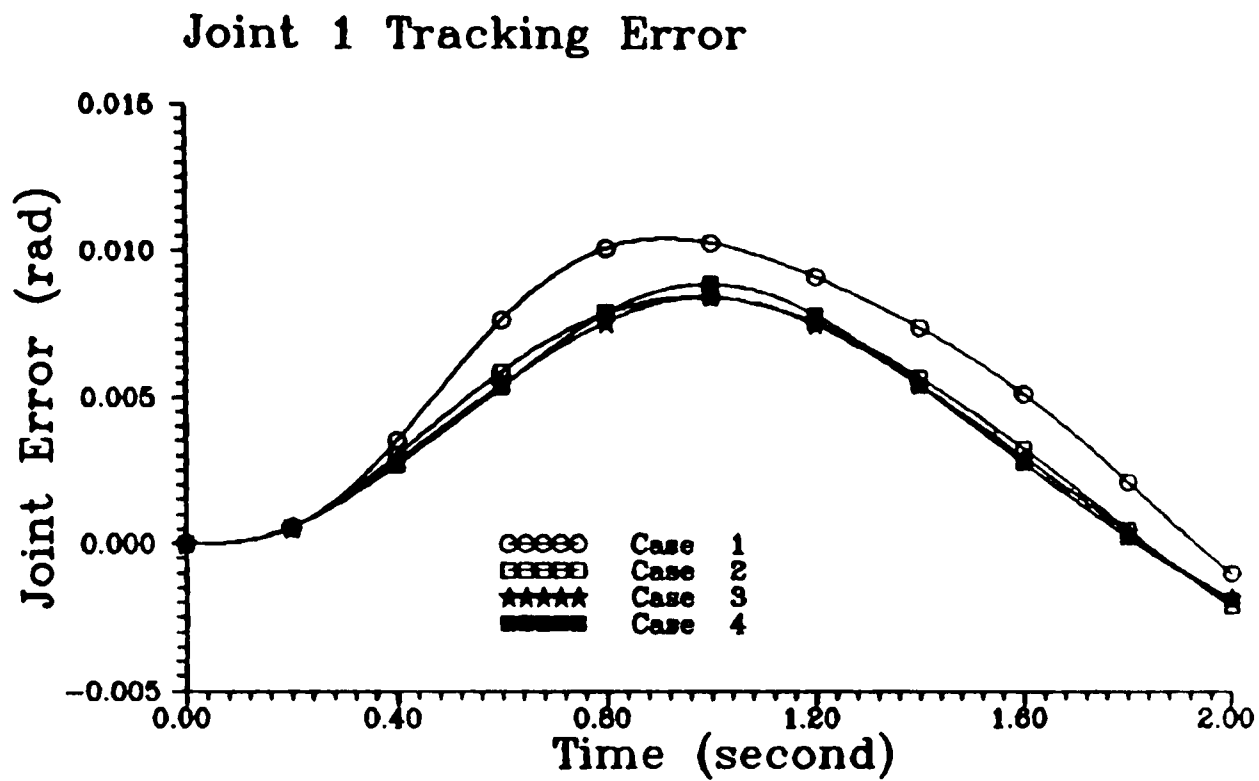


FIGURE 4.26 : Joint 1 Tracking Errors For 10 Kg.  
Load For Different  $Q_i$  Matrix

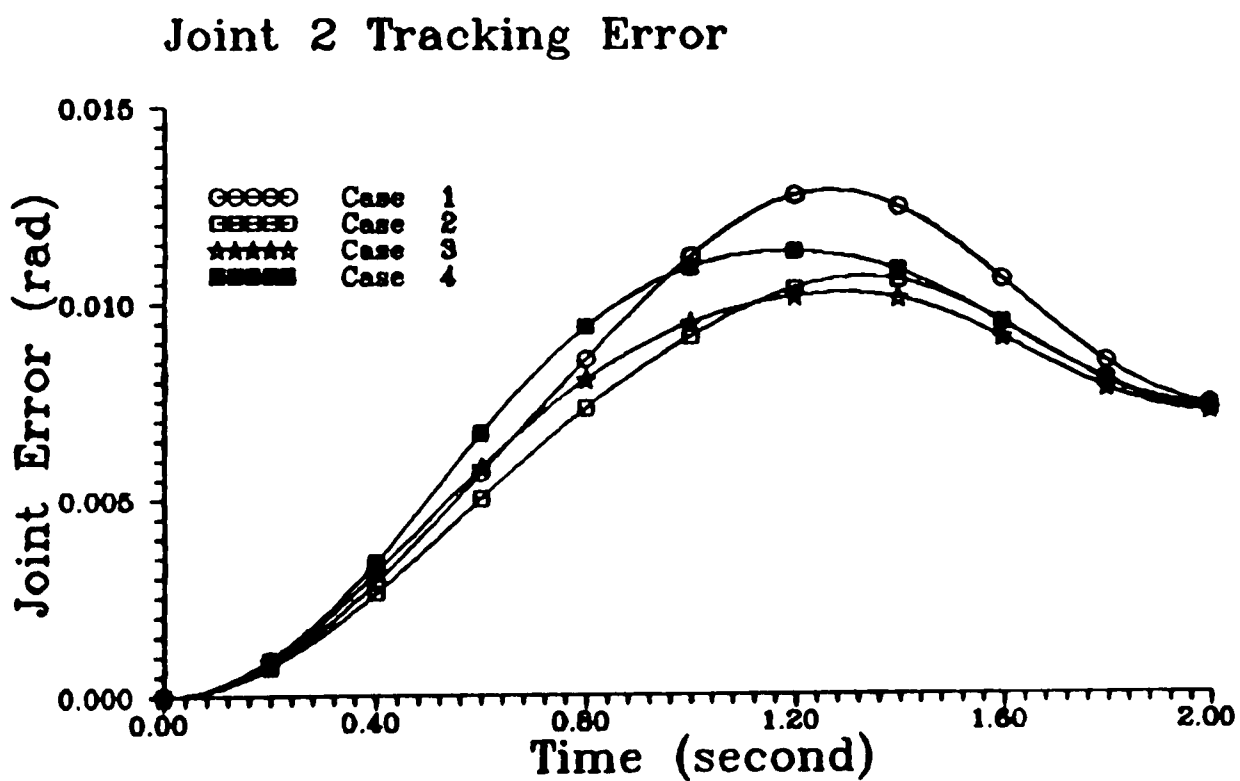


FIGURE 4.27 : Joint 2 Tracking Errors For 10 Kg.  
For Different  $Q_i$  Matrix

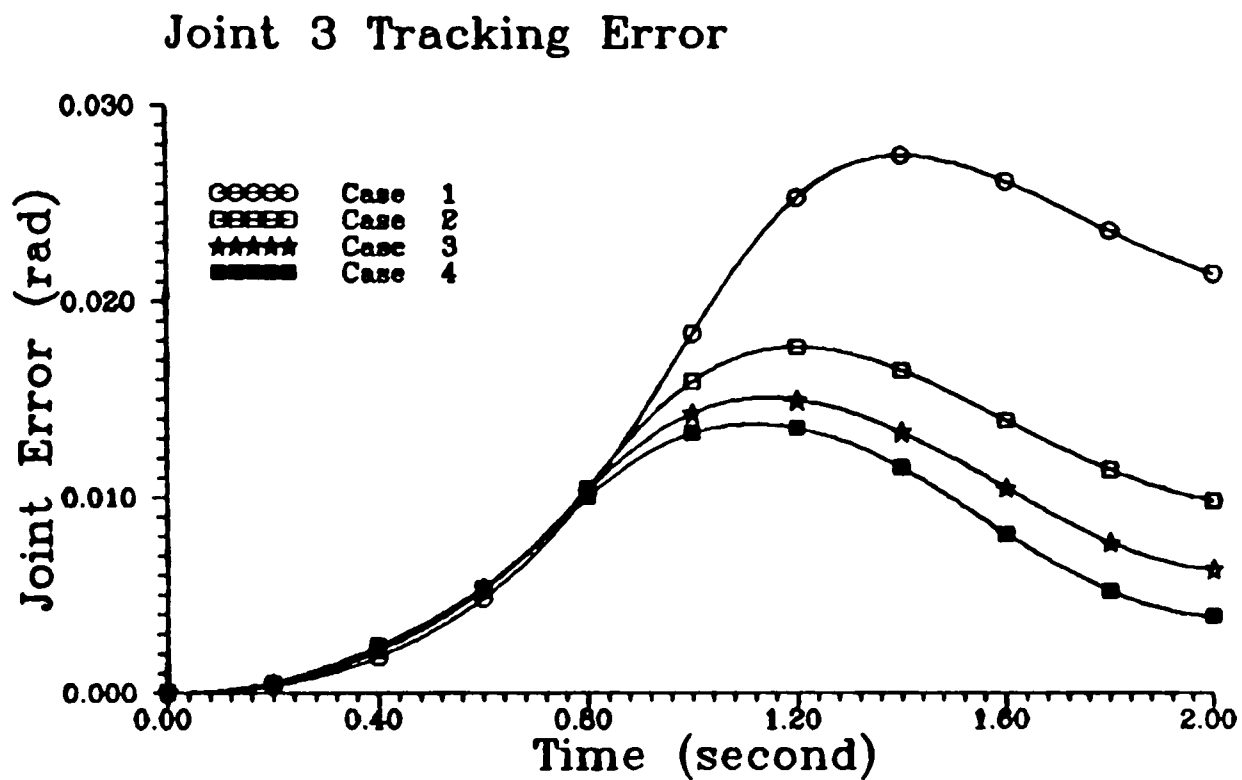


FIGURE 4.28 : Joint 3 Tracking Errors For 10 Kg. Load For Different  $Q_i$  Matrix

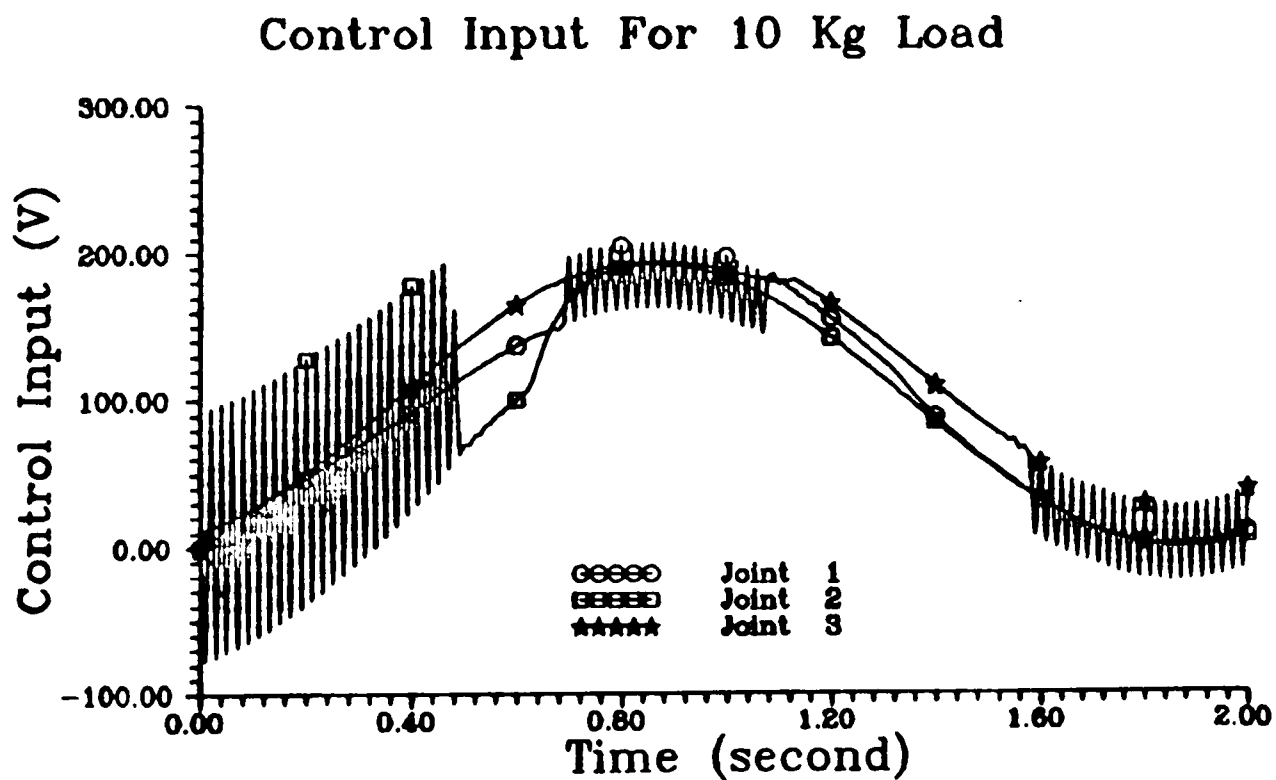


FIGURE 4.29 : Joint Control Inputs For 10 Kg. Load For  $Q_i = 1.1 I_3$

that as long as  $\|E_i(x,t)\|$  satisfies assumption (ii) for all  $i \in \mathfrak{I}$ , no matter how large the loads and the uncertainties are,\* an arbitrarily small set of  $\mathfrak{X}(\underline{\kappa})$  can be obtained by properly choosing  $\epsilon_i$  and  $Q_i$ .

## 4.6 CONCLUSION

Based on a deterministic approach, a framework for designing decentralized nonlinear feedback control law has been presented for tracking control of a robot manipulator. The system is treated as a set of interconnected subsystems with bounded uncertainties. The decentralized control approach utilizes only the local states and the bounds of the uncertainties as the feedback information, and satisfaction of a certain sufficient condition results in practical stability of the overall uncertain robot manipulator system despite the absence of information between the local subsystems. It has been shown theoretically and through simulations that the error between the response of the actual robotic system and that of the reference trajectory is uniformly ultimately bounded with respect to any arbitrarily small set of ultimate boundedness. That is the controller forces the robotic system to follow the reference trajectory within a close neighbourhood of the reference trajectory, even when substantial uncertainties and nonlinearities exist in the system. In other words, the controller is robust to any uncertainties and nonlinearities present in the system. The drawback of the approach is that the tracking precision is dependent on the magnitude of the interconnection functions. If the interconnections between the subsystems are strong, the uniform ultimate boundedness set will be large. Hence, the tracking error between the robot manipulator and the desired trajectory will increase. However, the ultimate boundedness set can be made smaller if the interconnection matrices are compensated directly. This will be presented in the following Chapter.

---

\* within the specified bounds.

## CHAPTER 5

# DECENTRALIZED GLOBAL TRACKING CONTROLLER DESIGN FOR ROBOT MANIPULATORS

### 5.1 INTRODUCTION

In the completely decentralized control method as presented in the previous chapter, the interconnections between the subsystems are completely neglected, yielding a loss of information about the behaviour of the interconnection and their dynamic effects on each subsystem. The radius of the uniform ultimate boundedness set is also dependent on the bound of the interconnection matrices  $\bar{g}_{ij}$ . Thus, if the the interconnections between the subsystems are strong, the uniform ultimate boundedness set will be large. Hence, the tracking error between the robot manipulator and the desired trajectory will increase. The uniform ultimate boundedness set can be made smaller if equation (4.63) is independent of the term  $\bar{g}_{ij}$ . This can be done if the interconnection matrices are compensated directly by using a decentralized global control method.

The use of a global control concept in a decentralized control strategy for a robot manipulator in order to overcome the discrepancy of the completely decentralized control method has been considered before. For example, Vukobratovic and Stokic [1983], and Stokic and Vukobratovic [1984] postulated that a global force feedback control law may be introduced in their decentralized control algorithm to compensate for the destabilizing influence of the couplings among the subsystems. However, the force feedback controller requires either the introduction of several force transducers which represents additional technical problems and increases the system costs, or the on-line computation of the coupling functions which is very time consuming and complex.

In this Chapter, a type of decentralized global nonlinear state feedback control law based on a deterministic approach is presented for the tracking control of a robot manipulator. The controller utilizes the local states as well as the states of the connecting subsystems as the feedback information. As with the method presented in the previous chapter, the controller is designed based only on the bounds of the nonlinearities, the uncertainties, and the interconnection functions present in the system. It is assumed that these bounds are known.

It will be shown theoretically and through computer simulations that the proposed controller will render the tracking errors between the robot manipulator responses and the corresponding reference trajectories uniformly ultimately bounded with respect to a small uniform ultimate boundedness set.

The following section outlines the formulation of the decentralized global tracking problem.

## 5.2 PROBLEM FORMULATION

The dynamic model of the robot manipulator in the form of equations (4.12) – (4.15) are being considered again. For the robot manipulator system as given by equation (4.12) :

$$\dot{X}_i(t) = [A_i + \Delta A_i(X, \xi, t)] X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t) , \quad i \in \mathcal{I} , \quad (5.1)$$

the objective is to design  $U_i(t)$  for each of the robot manipulator subsystem (each dof) based on its local states as well as the states of the connecting subsystems, such that, for any uncertainties and nonlinearities present in the system, the overall robot manipulator with the decentralized control law

$$U(t) = [ U_1(t) , U_2(t) , \dots , U_N(t) ]^T , \quad (5.2)$$

tracks the reference state trajectory  $X_d(t)$  as defined by equation (4.16), as closely as possible for all time  $t$ .

To synthesize the control algorithm, the following assumptions are required :

- i. The pair  $(A_i, B_i)$  is stabilizable,
- ii. There exist continuous functions  $H_i(X, \xi, t) \in \mathbb{R}^{1 \times 3}$ , and  $E_i(X, \xi, t) \in \mathbb{R}$ , such that for all  $X \in \mathbb{R}^N$  and all  $t$  :

$$\Delta A_i(X, \xi, t) = B_i H_i(X, \xi, t) \quad (5.3)$$

$$\Delta B_i(X, \xi, t) = B_i E_i(X, \xi, t) \quad (5.4)$$

$$\|E_i(X, \xi, t)\| < 1 . \quad (5.5)$$

- iii. There exist a Lebesgue function  $\Omega_i(t) \in \mathbb{R}$ , which is integrable on bounded intervals, such that

$$\dot{X}_{di}(t) = A_i X_{di}(t) + B_i \Omega_i(t) . \quad (5.6)$$

- iv. There exist continuous functions  $G_{ij}(X, \xi, t) \in \mathbb{R}^{1 \times 3}$ , such that for all  $X \in \mathbb{R}^N$  and all  $t$  :

$$A_{ij}(X, \xi, t) = B_i G_{ij}(X, \xi, t) . \quad (5.7)$$

Assumptions (i) – (iii) are similar to assumptions (i) to (iii) in the previous chapter and, hence, their explanation are omitted here.



In order to realize the decentralized global controller, assumption (iv) is needed to ensure that the interconnection functions  $A_{ij}(X, \xi, t)$  are within the range space of the input matrix  $B_i$ . Thus, the interconnection functions can be compensated by the control input  $U_i(t)$  which enters the  $i$ th subsystem through the input matrix  $B_i$ . The continuous functions  $G_{ij}(X, \xi, t)$  exist if the following rank condition is satisfied :

$$\text{rank} [ B_i ] = \text{rank} [ B_i, A_{ij}(X, \xi, t) ] . \quad (5.8)$$

If this assumption is not satisfied, the interconnection functions  $A_{ij}(X, \xi, t)$  cannot be compensated by the controller even if they are known. However, in view of equation (3.21), assumption (iv) always holds for a robot manipulator state space model derived with the joint angles, velocities and accelerations as the state variables.

It is also assumed here that there is no unstable fixed mode present in the system.

The proposed decentralized tracking control method based on a global approach is presented next.

### 5.3 DECENTRALIZED NONLINEAR TRACKING CONTROLLER DESIGN – GLOBAL APPROACH

Recall the error state equation (4.26) for the  $i$ th subsystem :

$$\dot{Z}_i(t) = A_i Z_i(t) + \Delta A_i(X, \xi, t) X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t) - B_i \Omega_i(t) . \quad (5.9)$$

From the given assumptions (i) to (iv), equation (5.9) becomes (the arguments have been omitted),

$$\dot{Z}_i(t) = A_i Z_i + B_i \left\{ H_i X_i - \Omega_i + [I + E_i] U_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j \right\}. \quad (5.10)$$

For each subsystem, a decentralized controller of the following form is considered:

$$U_i(t) \triangleq U_{Gi}(t) = \tilde{U}_i(Z_i, t) + \Phi_{Gi}(Z_i, \xi, t), \quad (5.11)$$

where  $\tilde{U}_i(Z_i, \xi, t)$  is as given in equation (4.31), and

$$\Phi_{Gi}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_{Gi}(Z_i, \xi, t)}{\|\mu_{Gi}(Z_i, \xi, t)\|} \rho_{Gi}(Z_i, \xi, t) & \text{if } \|\mu_{Gi}(Z_i, \xi, t)\| > \epsilon_i \\ \|\Phi_{Gi}(Z_i, \xi, t)\| \leq \rho_{Gi}(Z_i, \xi, t) & \text{if } \|\mu_{Gi}(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (5.12)$$

$$\mu_{Gi}(Z_i, \xi, t) = B_i^T P_i Z_i(t) \rho_{Gi}(Z_i, \xi, t) \quad (5.13)$$

$$\rho_{Gi}(Z_i, \xi, t) \triangleq \max_{\Delta b_i^i \in \mathcal{Y}} [1 - \|E_i(\Delta b_i^i)\|]^{-1} \left( \begin{aligned} &\max_{\Delta a_{ij}^i \in \mathcal{R}} \|H_i(\Delta a_{ij}^i) X_i(t)\| + \max_{\Delta b_i^i \in \mathcal{Y}} \|E_i(\Delta b_i^i) \tilde{U}_i(Z_i, \xi, t)\| + \\ &\max_{a_{ij}^{ij} \in \mathcal{V}} \left\| \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij}(a_{ij}^{ij}) X_j(t) \right\| \end{aligned} \right), \quad (5.14)$$

and, the positive definite matrix  $P_i$  and the closed loop system matrix  $\tilde{A}_i$ , are as defined in equations (4.36) and (4.35), respectively. Figure 5.1 illustrates the

proposed decentralized global tracking control structure.

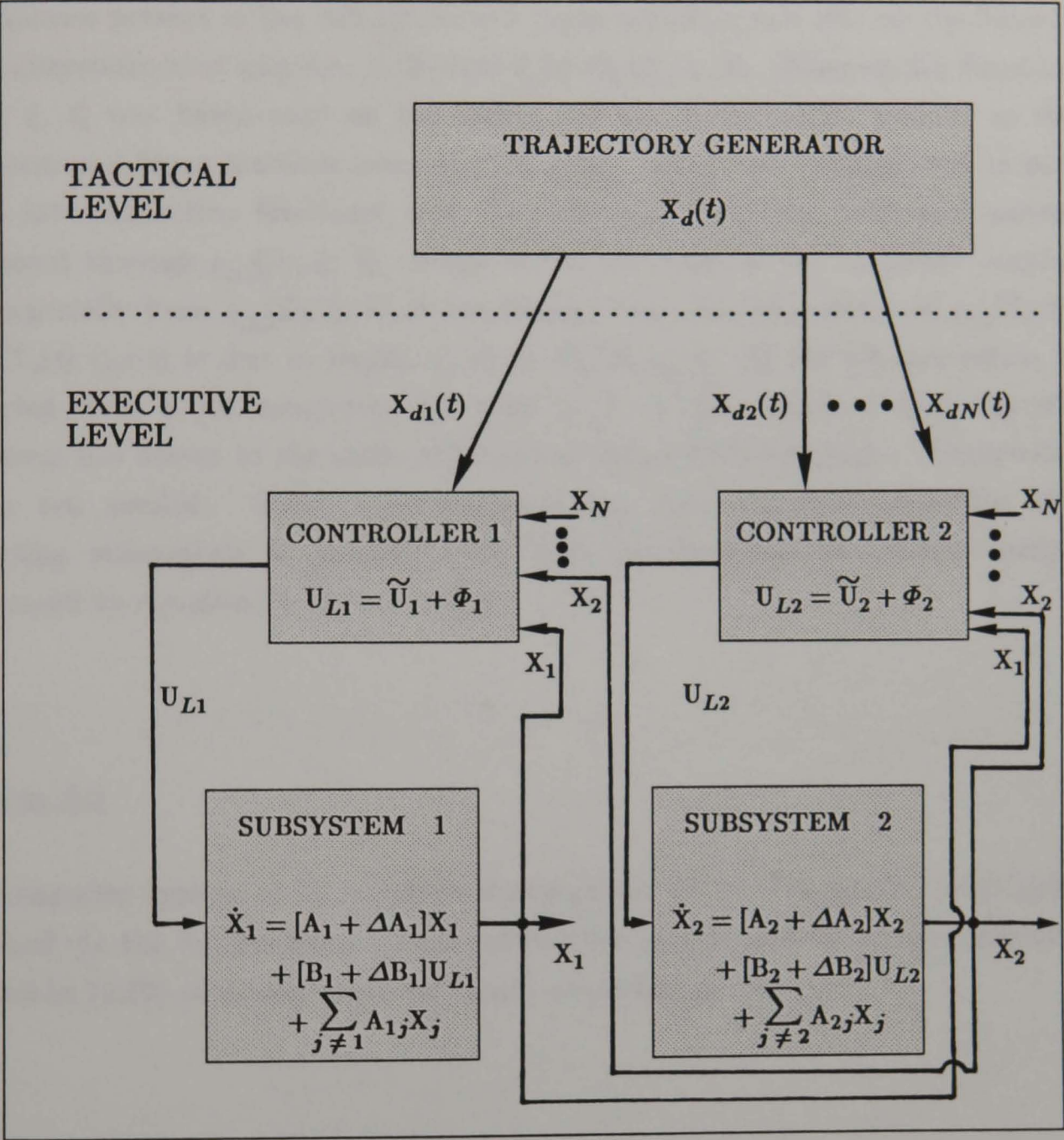


FIGURE 5.1 : Decentralized Control Of Robot Manipulator  
 – Global Approach

The main difference between the control law presented here and the decentralized local control law presented in the previous chapter is in the way the function  $\rho_{G_i}(Z_i, \xi, t)$  is formulated. It can be seen from equation (5.14) that  $\rho_{G_i}(Z_i, \xi, t)$  is formulated based not only on the possible bound of the uncertainties present in the subsystem and input matrices, but also on the bounds of the interconnection matrices represented by  $G_{ij}(X, \xi, t)$ . Whereas the function  $\rho_{L_i}(Z_i, \xi, t)$  was based only on the bounds of the uncertainties present in the subsystem and input matrices (see equation 4.34). Thus, the uncertainties as well as the interconnection functions, may be compensated by the nonlinear control component through  $\rho_{G_i}(Z_i, \xi, t)$ . From the construction of the nonlinear control part, especially from  $\rho_{G_i}(Z_i, \xi, t)$ , it can be seen that the dependence of  $\rho_{G_i}(Z_i, \xi, t)$  on  $X_j(t)$  ( $j \neq i$ ) is due to  $G_{ij}(X, \xi, t)$  or  $A_{ij}(X, \xi, t)$ . If the  $i$ th subsystem is connected with a  $j$ th subsystem for some  $j$ , it is then required that the  $i$ th subsystem has access to the state information of the  $j$ th subsystem. Otherwise, this is not needed. Here, it is assumed that local communication for the connecting subsystems is possible such that the decentralized control design represented by equation (5.11) is feasible.

### Theorem 5.1

The composite system (5.1) satisfying assumptions (i) to (iv) can be practically stabilized via the decentralized global control law (5.11), and tracks the reference trajectories (4.16) to within any neighbourhood of  $\mathfrak{B}(\eta_G)$ , where

$$\eta_G = \sqrt{\frac{\sum_i^N 4\epsilon_i}{\lambda_{\min}(\mathbb{T}_G)}} \quad . \quad (5.15)$$

### Proof

By substituting equation (5.11) into (5.10), gives

$$\begin{aligned}
 \dot{Z}_i(t) &= A_i Z_i - B_i \Omega_i + B_i [H_i X_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j] + B_i [I + E_i] [\tilde{U}_i + \Phi_{Gi}] \\
 &= \tilde{A}_i Z_i - B_i [K_i Z_i + \Omega_i] + B_i [H_i X_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j] + B_i [I + E_i] [\tilde{U}_i + \Phi_{Gi}] \\
 &= \tilde{A}_i Z_i + B_i [H_i X_i + E_i \tilde{U}_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j] + B_i [I + E_i] \Phi_{Gi} \\
 &= \tilde{A}_i Z_i + B_i e_{Gi} + B_i \Phi_{Gi},
 \end{aligned} \tag{5.16}$$

where

$$e_{Gi} = H_i X_i + E_i \tilde{U}_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j + E_i \Phi_{Gi}. \tag{5.17}$$

Let the Lyapunov function for each subsystem be

$$\mathcal{L}_i(Z, t) = Z_i^T P_i Z_i, \tag{5.18}$$

and the composite Lyapunov function for the overall system be

$$\mathcal{L}(Z, t) = \sum_{i=1}^N Z_i^T P_i Z_i. \tag{5.19}$$

Beginning with the composite Lyapunov equation and equation (5.16), the proof of the theorem can be accomplished by following the same procedure as outlined in section 4.4. From equations (5.16) and (5.19), the derivative of the Lyapunov equation can be obtained as follows :

$$\dot{\mathcal{L}}(Z, t) = \sum_i^N 2Z_i^T P_i \tilde{A}_i Z_i + \sum_i^N 2Z_i^T P_i B_i \{ \Phi_{Gi} + e_{Gi} \}$$

$$= \sum_{\mathbf{i}}^N 2\mathbf{Z}_{\mathbf{i}}^T \mathbf{P}_{\mathbf{i}} \tilde{\mathbf{A}}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}} + \sum_{\mathbf{i}}^N 2(\mathbf{B}_{\mathbf{i}}^T \mathbf{P}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}})^T \{ \Phi_{G_{\mathbf{i}}} + \mathbf{e}_{G_{\mathbf{i}}} \} . \quad (5.20)$$

From Rayleigh's principle [Franklin, 1968] :

$$\lambda_{\min}(\mathbf{Q}_{\mathbf{i}}) \|\mathbf{Z}_{\mathbf{i}}\|^2 \leq \mathbf{Z}_{\mathbf{i}}^T \mathbf{Q}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}} \leq \lambda_{\max}(\mathbf{Q}_{\mathbf{i}}) \|\mathbf{Z}_{\mathbf{i}}\|^2 , \quad (5.21)$$

and in view of equation (4.45), the first term on the right-hand-side (RHS) of equation (5.20) can be written as follows :

$$\sum_{\mathbf{i}}^N 2\mathbf{Z}_{\mathbf{i}}^T \mathbf{P}_{\mathbf{i}} \tilde{\mathbf{A}}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}} \leq - \sum_{\mathbf{i}}^N \lambda_{\min}(\mathbf{Q}_{\mathbf{i}}) \|\mathbf{Z}_{\mathbf{i}}\|^2 . \quad (5.22)$$

From the fact that  $\|\Phi_{G_{\mathbf{i}}}\| \leq \rho_{G_{\mathbf{i}}}$  (equation 5.12) , implies,

$$\|\mathbf{e}_{G_{\mathbf{i}}}\| = \left\| \mathbf{H}_{\mathbf{i}} \mathbf{X}_{\mathbf{i}} + \mathbf{E}_{\mathbf{i}} \Phi_{G_{\mathbf{i}}} + \sum_{\substack{j=1 \\ j \neq \mathbf{i}}}^N \mathbf{G}_{\mathbf{i}j} \mathbf{X}_j + \mathbf{E}_{\mathbf{i}} \tilde{\mathbf{U}}_{\mathbf{i}} \right\| \leq \rho_{G_{\mathbf{i}}} . \quad (5.23)$$

In deriving equation (5.23), equation (5.14) has been utilized. For the last term on the RHS of equation (5.20) :

i. If  $\mu_{G_{\mathbf{i}}} > \epsilon_{\mathbf{i}}$  , implies

$$\Phi_{G_{\mathbf{i}}} = - \frac{\mathbf{B}_{\mathbf{i}}^T \mathbf{P}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}}}{\|\mathbf{B}_{\mathbf{i}}^T \mathbf{P}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}}\|} \rho_{G_{\mathbf{i}}} , \quad (5.24)$$

which gives

$$\sum_{\mathbf{i}}^N 2(\mathbf{B}_{\mathbf{i}}^T \mathbf{P}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}})^T \{ \Phi_{G_{\mathbf{i}}} + \mathbf{e}_{G_{\mathbf{i}}} \} \leq \sum_{\mathbf{i}}^N 2 \|\mathbf{B}_{\mathbf{i}}^T \mathbf{P}_{\mathbf{i}} \mathbf{Z}_{\mathbf{i}}\| \|\Phi_{G_{\mathbf{i}}} + \mathbf{e}_{G_{\mathbf{i}}}\|$$

$$\begin{aligned}
&\leq \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \{ \|\Phi_{G_i}\| + \|e_{G_i}\| \} \\
&\leq - \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \rho_{G_i} + \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \rho_{G_i} \\
&= 0 ;
\end{aligned} \tag{5.25}$$

ii. If  $\mu_{G_i} \leq \epsilon_i$ , hence,

$$\|B_i^T P_i Z_i\| \rho_{G_i} \leq \epsilon_i , \tag{5.26}$$

then,

$$\begin{aligned}
\sum_{i=1}^N 2(B_i^T P_i Z_i)^T \{\Phi_{G_i} + e_{G_i}\} &\leq \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \|\Phi_{G_i} + e_{G_i}\| \\
&\leq \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \|\Phi_{G_i}\| + \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \|e_{G_i}\| \\
&\leq \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \rho_{G_i} + \sum_{i=1}^N 2 \|B_i^T P_i Z_i\| \rho_{G_i} \\
&\leq \sum_{i=1}^N 4 \|B_i^T P_i Z_i\| \rho_{G_i} \\
&\leq \sum_{i=1}^N 4 \epsilon_i .
\end{aligned} \tag{5.27}$$

From equations (5.22), (5.25), and (5.27), the derivative of the Lyapunov function along the solution of (5.16) is given as follows :

$$\dot{L}(Z, t) \leq - \sum_{j=1}^N \lambda_{\min}(Q_j) \|Z_j\|^2 + \sum_{j=1}^N 4\epsilon_j \tag{5.28}$$

Using (4.57) and (4.58), the above equation can be rewritten in the following form :

$$\begin{aligned}\dot{\mathbf{z}}(\mathbf{Z}, t) &\leq - \tilde{\mathbf{Z}}^T \mathbb{T}_G \tilde{\mathbf{Z}} + \sum_i^N 4 \epsilon_i \\ &\leq - \lambda_{\min}(\mathbb{T}_G) \|\tilde{\mathbf{Z}}\|^2 + \sum_i^N 4 \epsilon_i ,\end{aligned}\quad (5.29)$$

where, the vector  $\tilde{\mathbf{Z}}$  is as defined by equation (4.57) and,  $\mathbb{T}_G$  is an  $N \times N$  test matrix whose elements are:

$$\mathbb{T}_{Gij} = \begin{cases} \lambda_{\min}(Q_i) & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} . \quad (5.30)$$

Since the matrix  $\mathbb{T}_G$  is a diagonal matrix and  $Q_i > 0$  for each  $i \in \mathfrak{I}$ ,  $\mathbb{T}_G$  is positive definite. Consequently,  $\dot{\mathbf{z}} < 0$  for all  $t \in \mathfrak{R}$  and all  $\tilde{\mathbf{Z}}(t) \in \mathfrak{B}^c(\eta_G)$ , where  $\mathfrak{B}^c(\eta_G)$  is the complement of the closed ball  $\mathfrak{B}(\eta_G)$ , centred at  $\tilde{\mathbf{Z}} = 0$  with radius

$$\eta_G = \sqrt{\frac{\sum_i^N 4\epsilon_i}{\lambda_{\min}(\mathbb{T}_G)}} , \quad (5.31)$$

and the error system (5.9) is uniformly ultimately bounded with respect to the set  $\mathfrak{X}(\underline{\kappa})$ , where

$$\mathfrak{X}(\underline{\kappa}) \triangleq \{ \tilde{\mathbf{Z}} \in \mathfrak{R}^{3N} \mid \tilde{\mathbf{Z}}^T \mathbf{P} \tilde{\mathbf{Z}} \leq \underline{\kappa} \} , \quad (5.32)$$



$$\underline{\kappa} = \lambda_{\max}(P) \eta_G^2 \quad (5.33)$$

$$P = \text{diag} [ P_1, P_2, \dots, P_N ], \text{ and } P_i > 0. \quad (5.34)$$

The set  $\mathfrak{X}(\underline{\kappa})$  is the smallest Lyapunov ellipsoid that contains the closed ball  $\mathfrak{B}(\eta_G)$ . Then, every solution  $Z(t)$  of (5.9) must be uniformly ultimately bounded within every  $\mathfrak{X}(\underline{\kappa})$ . In other words, system (5.1) with the decentralized global control law (5.11) has existence, continuation of solution  $X(t)$  and tracks  $X_d(t)$  to within any neighbourhood of  $\mathfrak{B}(\eta_G)$ . The set  $\mathfrak{B}(\eta_G)$  can be made arbitrarily small by properly choosing  $Q_i > 0$  and  $\epsilon_i > 0$ ,  $i = 1, 2, \dots, N$ . Thus, the error between the response of the actual system and the desired trajectory can be made arbitrarily small after a finite interval of time. This conclude the proof.  $\square$

It can be seen from equations (5.31)–(5.33) that the uniform ultimate boundedness set depends on  $\epsilon_i$ . Thus, in view of equation (5.31), the set can be made small by decreasing  $\epsilon_i$ . By comparing equation  $\eta_G$  in (5.31) and  $\eta_L$  in (4.62), it can be concluded that for a given  $\epsilon_i$  ( $i \in \mathfrak{I}$ ),  $\eta_G < \eta_L$  and, hence the uniform ultimate boundedness set for the integrated robot manipulator system with a decentralized local control law is larger than the uniform ultimate boundedness set of the system using a decentralized global control law. Thus, the tracking errors between the responses of the actual robot system using the decentralized global controller (5.11) and the desired trajectories are smaller than that of the system utilizing the decentralized local controller (4.30). The decrease in the size of the uniform ultimate boundedness set for the decentralized global control law is due to the fact that the nonlinear interconnection functions  $A_{ij}(X, \xi, t)$  have been taken into account and duly compensated by the decentralized global control approach; whereas in the decentralized local control approach, their influence have completely been ignored.

Another advantage of the decentralized global controller (5.11) is that the stability of the overall system can always be assured without referring to the test matrix  $\mathbb{T}_G$  since by design,  $Q_i$  is always chosen to be a positive definite matrix for each subsystems. Thus,  $\lambda_{\min}(\mathbb{T})$  is always positive.

In the following section, the effectiveness of the proposed approach is demonstrated by means of a computer simulation study.

## 5.4 SIMULATION AND RESULTS

The proposed decentralized global controller has been applied to the three dof robot manipulator actuated by armature controlled DC motors as considered in the previous chapters, and simulated on a digital computer.

Based on the known nominal matrices  $A_i$  and  $B_i$ , and the known maximum bounds on the uncertain elements of the matrices  $\Delta A_i(X, \xi, t)$ ,  $\Delta B_i(X, \xi, t)$ , and  $A_{ij}(X, \xi, t)$ , the control law for each subsystem was established according to equations (5.11) to (5.14). The elements of the matrices  $A_i$  and  $B_i$ , as well as the bounds on the elements of the matrices  $\Delta A_i(X, \xi, t)$ ,  $\Delta B_i(X, \xi, t)$  and  $A_{ij}(X, \xi, t)$  are as given by equation (3.26) in Chapter 3.

The continuous functions  $H_i(X, \xi, t)$  and  $E_i(X, \xi, t)$  in equation (5.14) are as computed in Chapter 4 (equations 4.79, 4.80 and 4.81). Thus, assumption (ii) is satisfied. The continuous functions  $G_{ij}(X, \xi, t)$  corresponding to the interconnection matrices  $A_{ij}(X, \xi, t)$  can be computed as follows :

$$\begin{aligned}
 G_{12}(X, \xi, t) &= 1.4313 \begin{bmatrix} 0 & a_{32}^{12} & a_{33}^{12} \end{bmatrix} \\
 G_{13}(X, \xi, t) &= 1.4313 \begin{bmatrix} 0 & a_{32}^{13} & a_{33}^{13} \end{bmatrix} \\
 G_{21}(X, \xi, t) &= 1.6004 \begin{bmatrix} 0 & a_{32}^{21} & a_{33}^{21} \end{bmatrix} \\
 G_{23}(X, \xi, t) &= 1.6004 \begin{bmatrix} a_{31}^{23} & a_{32}^{23} & a_{33}^{23} \end{bmatrix} \\
 G_{31}(X, \xi, t) &= 1.5751 \begin{bmatrix} 0 & a_{32}^{31} & a_{33}^{31} \end{bmatrix} \\
 G_{32}(X, \xi, t) &= 1.5751 \begin{bmatrix} a_{31}^{32} & a_{32}^{32} & a_{33}^{32} \end{bmatrix} .
 \end{aligned} \tag{5.35}$$

For the nonlinear part of the controller, it is sufficient to compute the functions  $\rho_{G_i}(Z_i, \xi, t)$  in equation (5.14) based on the maximum bound of the elements  $\Delta a_{ij}^i$ ,  $\Delta b_i^i$ , and  $a_{ij}^{ij}$ . Thus, the functions  $\rho_{G_i}(Z_i, \xi, t)$  may be given as follows :

$$\begin{aligned}
\rho_{G1}(Z_1, \xi, t) &= 1.0339 \left\{ \|4.1174 X_2^1 + 0.5271 X_3^1\| + \|0.0328 w_1\| + \right. \\
&\quad \|2.2809 X_2^2 + 0.1964 X_3^2\| + \\
&\quad \left. \|1.3024 X_2^3 + 0.1153 X_3^3\| \right\} \\
\rho_{G2}(Z_2, \xi, t) &= 1.0256 \left\{ \|28.1056 X_1^2 + 3.9371 X_2^2 + 0.1489 X_3^2\| + \right. \\
&\quad \|0.0249 w_2\| + \|0.9637 X_2^1 + 0.1687 X_3^1\| + (5.36) \\
&\quad \left. \|0.2816 X_1^3 + 3.9323 X_2^3 + 0.2094 X_3^3\| \right\} \\
\rho_{G3}(Z_3, \xi, t) &= 1.0003 \left\{ \|12.5947 X_1^3 + 0.3616 X_2^3 + 2.6029 \times 10^{-3} X_3^3\| + \right. \\
&\quad \|2.6154 \times 10^{-4} w_3\| + \|0.579 X_2^1 + 0.1016 X_3^1\| + \\
&\quad \left. \|0.6358 X_1^2 + 3.5252 X_2^2 + 0.0838 X_3^2\| \right\}.
\end{aligned}$$

The following form of the nonlinear controller (equation 5.12) law has been used :

$$\Phi_{Gi}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_{Gi}(Z_i, \xi, t)}{\|\mu_{Gi}(Z_i, \xi, t)\|} \rho_{Gi}(Z_i, \xi, t) & \text{if } \|\mu_{Gi}(Z_i, \xi, t)\| > \epsilon_i \\ -\frac{\mu_{Gi}(Z_i, \xi, t)}{\epsilon_i} \rho_{Gi}(Z_i, \xi, t) & \text{if } \|\mu_{Gi}(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (5.37)$$

where  $\mu_{Gi}(Z_i, \xi, t)$  is as given by equation (5.13) for  $i=1, 2, 3$ .

The linear part of the controller  $\tilde{U}_i(Z_i, t)$  in equation (5.11) is as given by equation (4.31) :

$$\tilde{U}_i(Z_i, t) = K_i Z_i(t) + \Omega_i(t), \quad i = 1, 2, 3. \quad (5.38)$$

The same functions  $\Omega_i(t)$  as given in Chapter 4 are used :

$$\begin{aligned}\Omega_1(t) &= 125.6744 \dot{\theta}_{d1}(t) + 15.6788 \ddot{\theta}_{d1}(t) + 1.4313 \dddot{\theta}_{d1}(t) \\ \Omega_2(t) &= 141.0512 \dot{\theta}_{d2}(t) + 17.5313 \ddot{\theta}_{d2}(t) + 1.6004 \dddot{\theta}_{d2}(t) \\ \Omega_3(t) &= 141.3782 \dot{\theta}_{d3}(t) + 17.2894 \ddot{\theta}_{d3}(t) + 1.5751 \dddot{\theta}_{d3}(t) .\end{aligned}\tag{5.39}$$

The same feedback gains as in equation (4.76) were used for each sub system :

$$\begin{aligned}\text{Subsystem 1} : \quad K_1 &= \begin{bmatrix} -1.145 & 119.7204 & 7.3775 \end{bmatrix} \\ \text{Subsystem 2} : \quad K_2 &= \begin{bmatrix} -400.0979 & -58.9978 & -14.4766 \end{bmatrix} \\ \text{Subsystem 3} : \quad K_3 &= \begin{bmatrix} -0.7089 & 136.5112 & 8.4689 \end{bmatrix} .\end{aligned}\tag{5.40}$$

Similarly, the positive definite symmetrical matrix  $Q_i$  was taken to be a 3x3 identity matrix for each subsystem. Thus, the corresponding solution of the matrix Lyapunov equation remained the same as in the previous chapter (equation 4.78).

The above controller and the three dof robot manipulator were simulated on a digital computer using a sampling period of 10 ms. In the simulations study, the Runge-Kutta-Butcher integration method and double precision arithmetic have been used. In evaluating the performance of the decentralized global controller, various simulations have been performed, and the results will be presented in the following subsections.

### 5.4.1 Comparison Between Local And Global Approach

One of the aims of the simulation is to compare the performance of the decentralized global control law with the decentralized local control approach. For that purpose, the control laws were computed under equal conditions and both systems were implemented to track the same desired trajectory.

Figures 5.2, 5.3 and 5.4 illustrate the position tracking responses for joint 1, 2 and 3, respectively, for the two control methods. In the simulation, the robot manipulator was required to track a desired trajectory starting from an initial position of  $\theta(0) = [ -0.8 , -1.5 , -0.5 ]^T$  radians and end at the final position of  $\theta(\tau) = [ 1.0 , 0.2 , 1.2 ]^T$  radians in  $\tau = 2$  seconds while carrying a 10Kg. load. For both controllers, the following values of  $\epsilon_i$  have been used :

$$\epsilon_1 = 1.1 \ ; \ \epsilon_2 = 0.5 \ ; \ \epsilon_3 = 0.9 \ .$$

For the decentralized global controller and the above values, the radius of the closed ball which is wholly contain inside the smallest Lyapunov ellipsoid can be computed as :

$$\eta_G = \sqrt{\frac{\sum_i^N 4\epsilon_i}{\lambda_{\min}(\mathbb{T}_G)}} = 3.1623 \ . \quad (5.41)$$

The corresponding joint velocity and joint acceleration tracking responses, together with the control inputs to the joint actuators for the decentralized global control method are as depicted in Figures 5.5, 5.6, and 5.7, respectively. These figures can be compared to Figures 4.8, 4.9, and 4.10, respectively, for the decentralized local control approach. Notice that the control signal for each joint actuator is smooth and continuous, and is within the operating limit of the actuators ( $\pm 240V$ ).

The figures show that both of the control approaches are capable of forcing the uncertain nonlinear robot manipulator to track a desired trajectory satisfactorily. However, it is obvious from Figures 5.2b, 5.3b, and 5.4b that, under the same conditions, the performance of the robot manipulator is far better

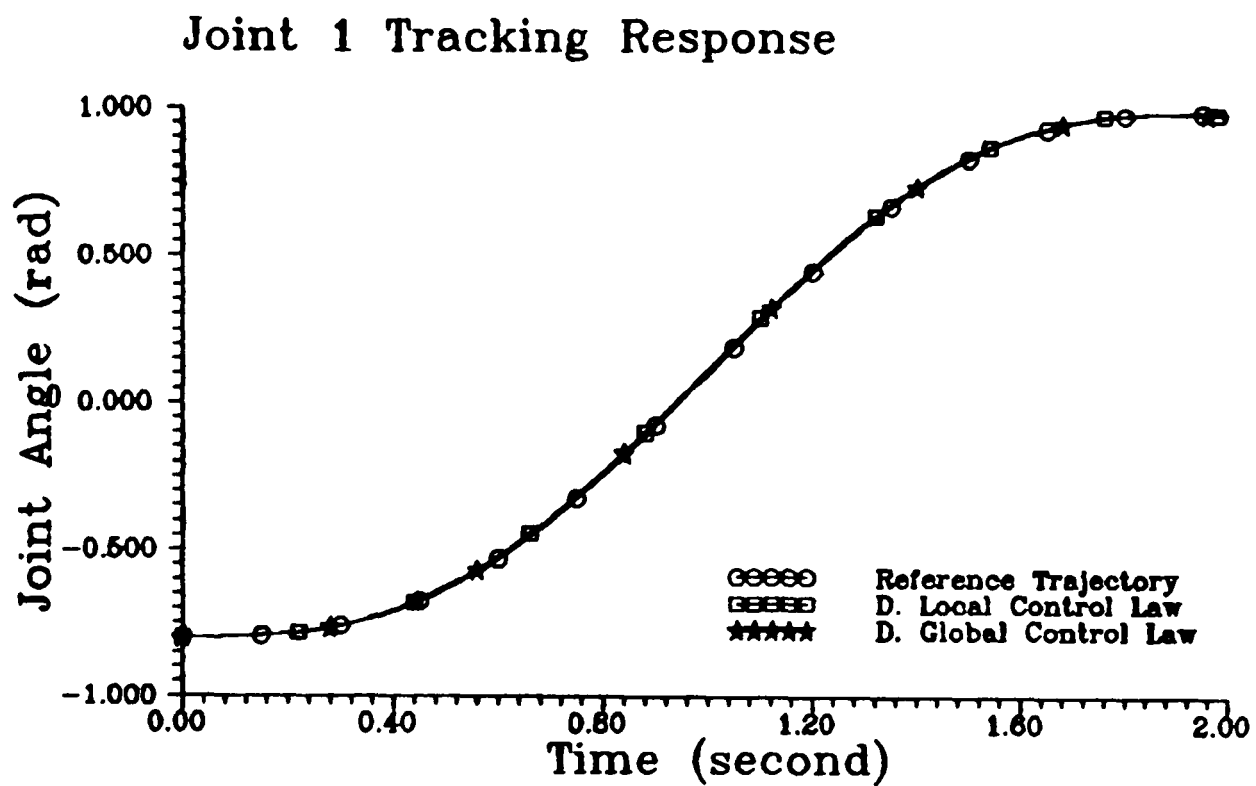


FIGURE 5.2 a : Joint 1 Tracking Response Under Decentralized Local and Global Control Methods

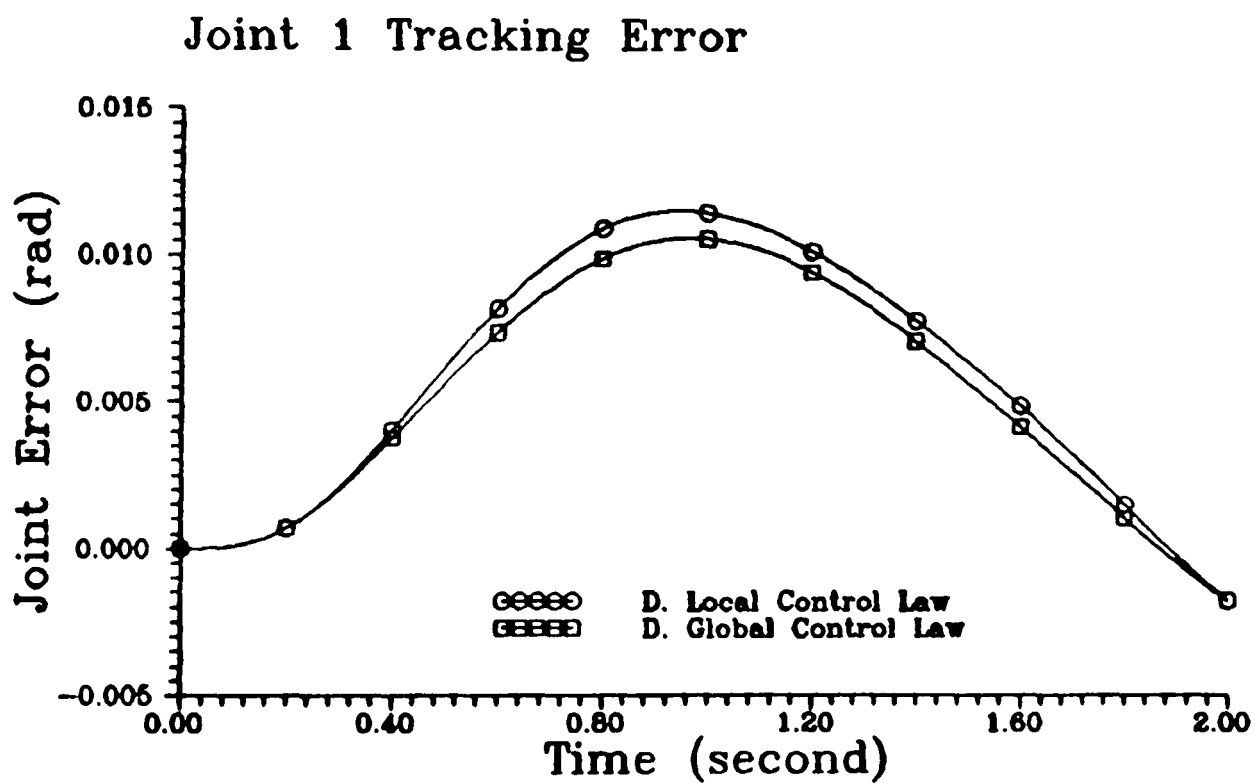


FIGURE 5.2 b : Joint 1 Tracking Error Under Decentralized Local and Global Control Methods

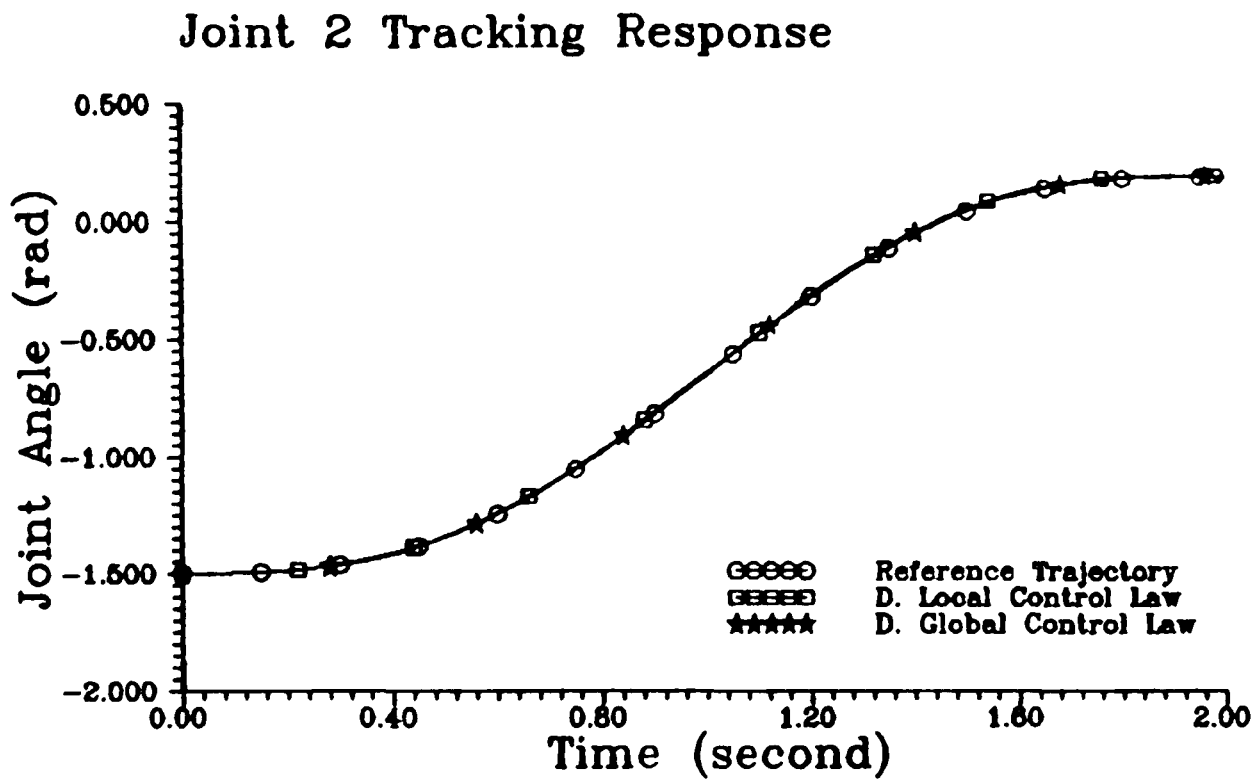


FIGURE 5.3 a : Joint 2 Tracking Response Under Decentralized Local and Global Control Methods

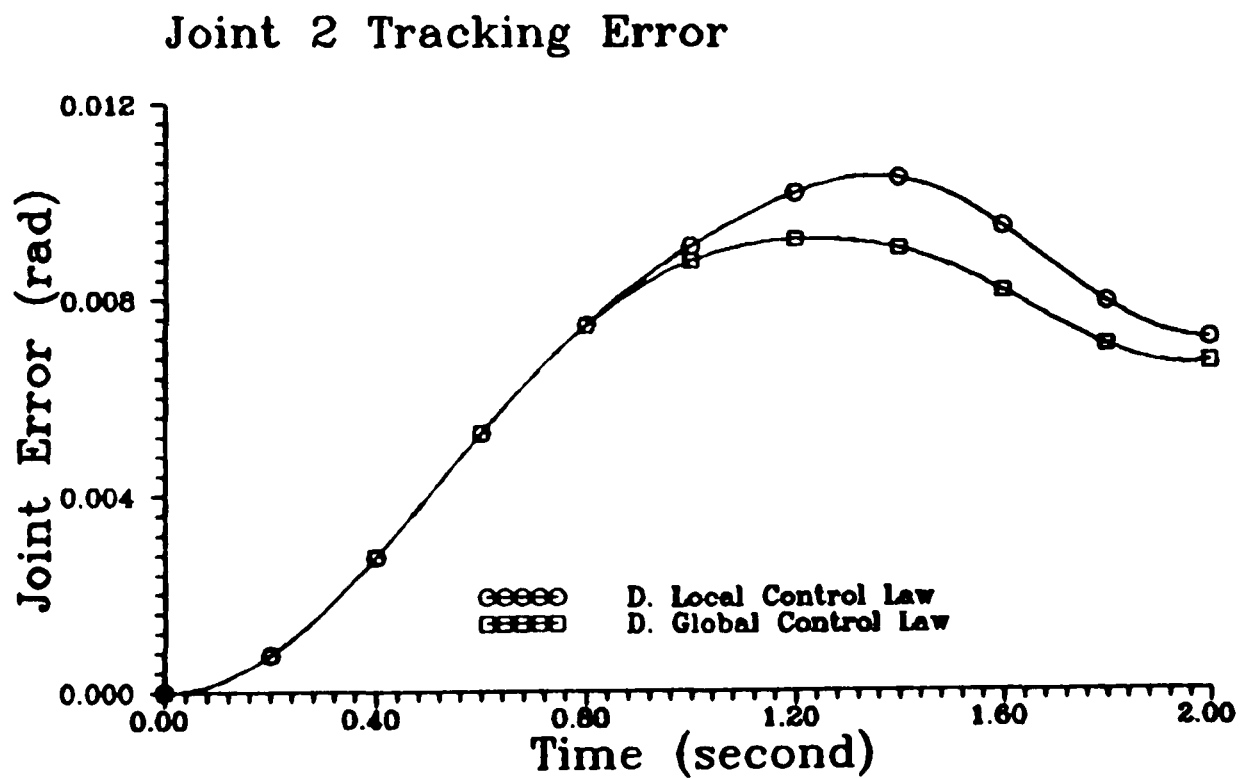


FIGURE 5.3b : Joint 2 Tracking Error Under Decentralized Local and Global Control Methods

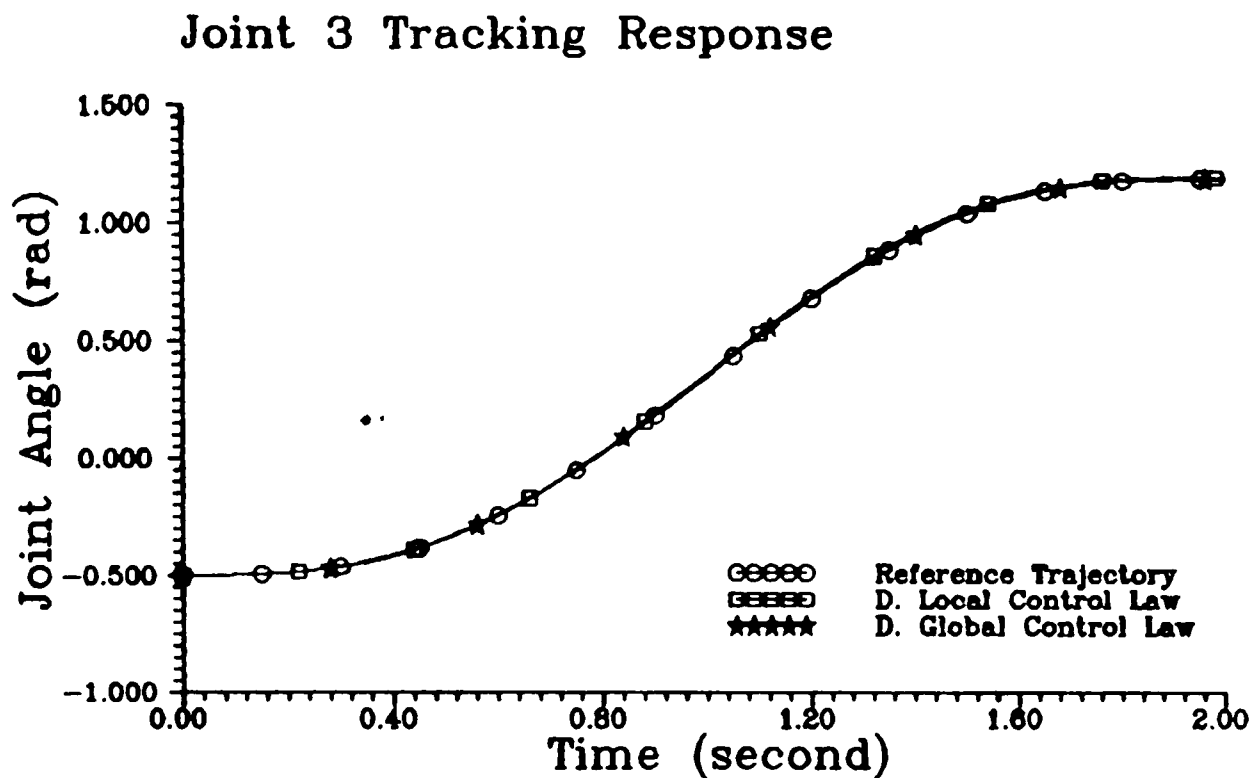


FIGURE 5.4 a : Joint 3 Tracking Response Under Decentralized Local and Global Control Methods

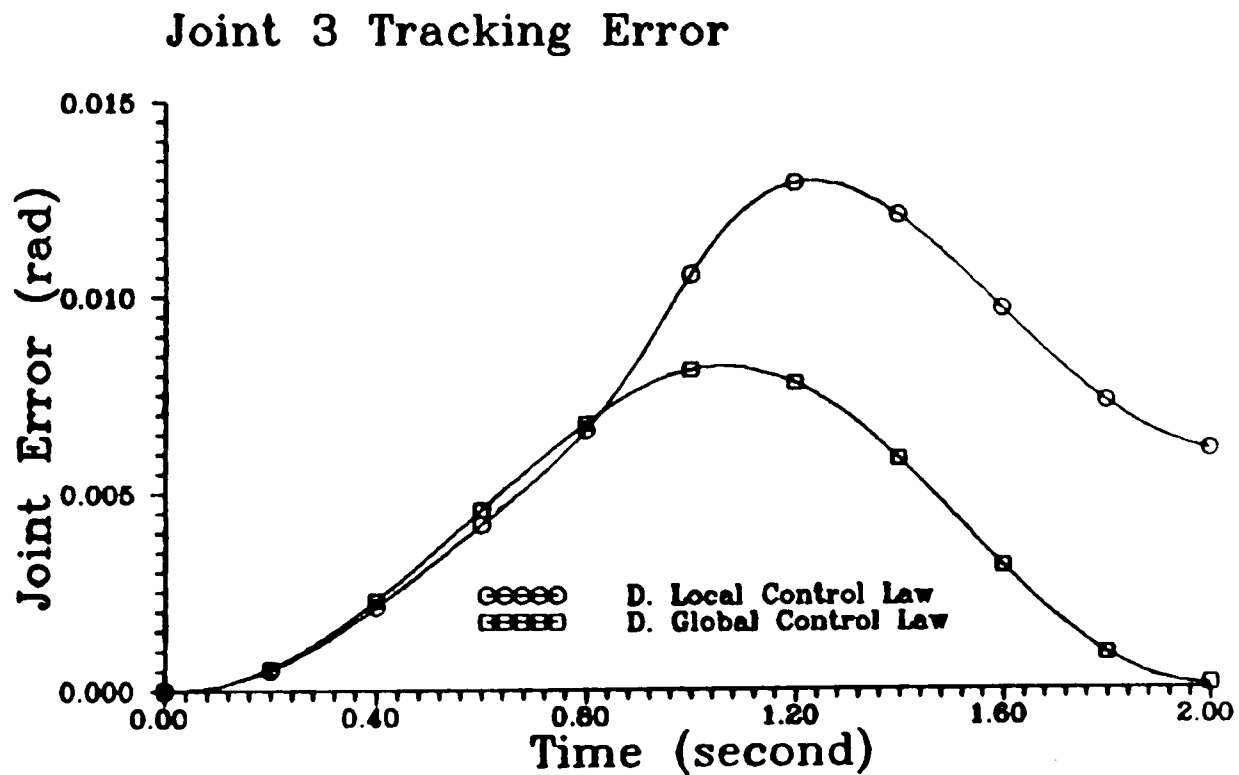


FIGURE 5.4 b : Joint 3 Tracking Error Under Decentralized Local and Global Control Methods



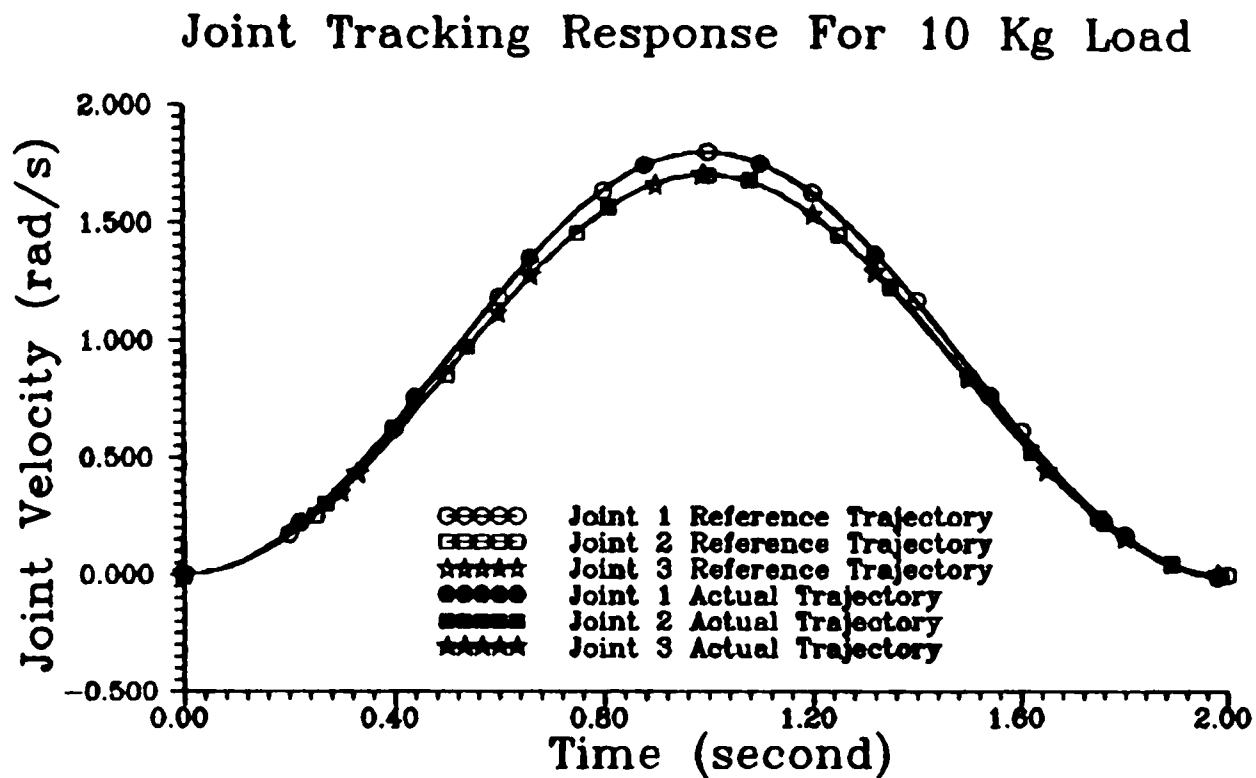


FIGURE 5.5 : Velocity Tracking Responses For 10 Kg. Load Using Decentralized Global Controller

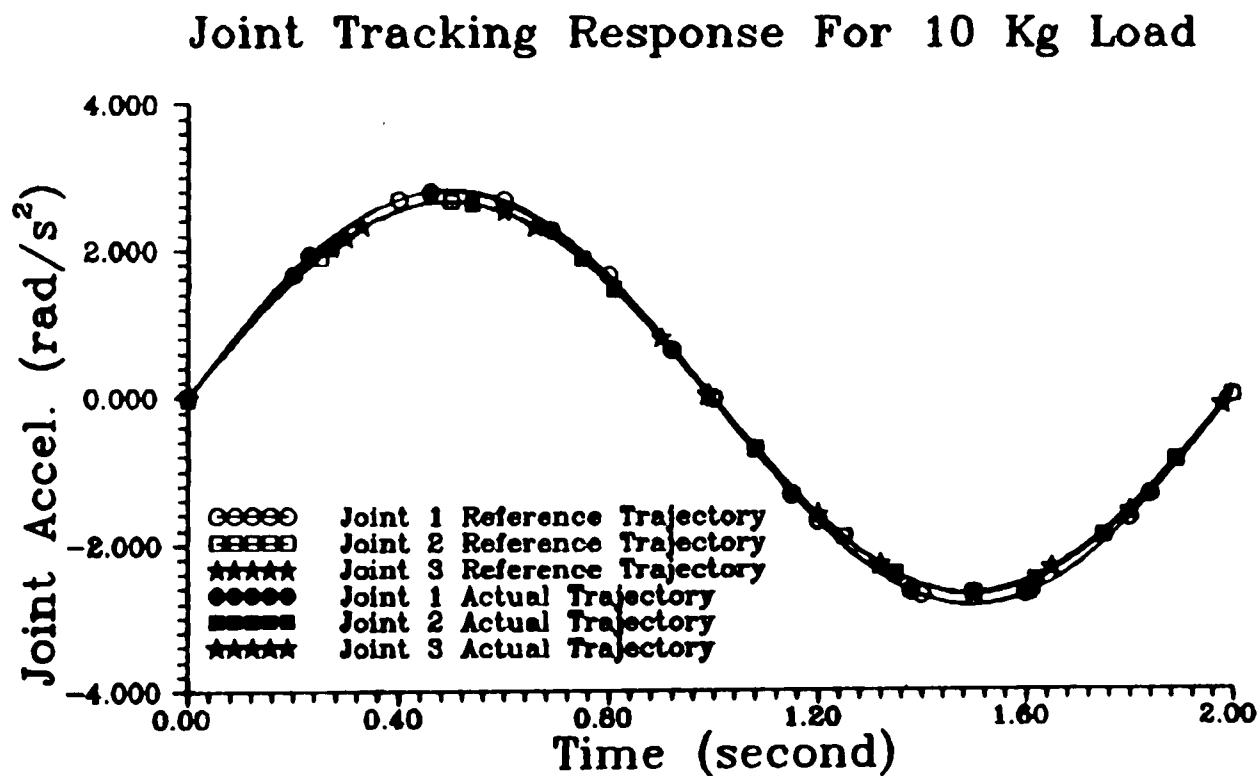


FIGURE 5.6 : Acceleration Tracking Responses For 10 Kg. Load Using Decentralized Global Control Method

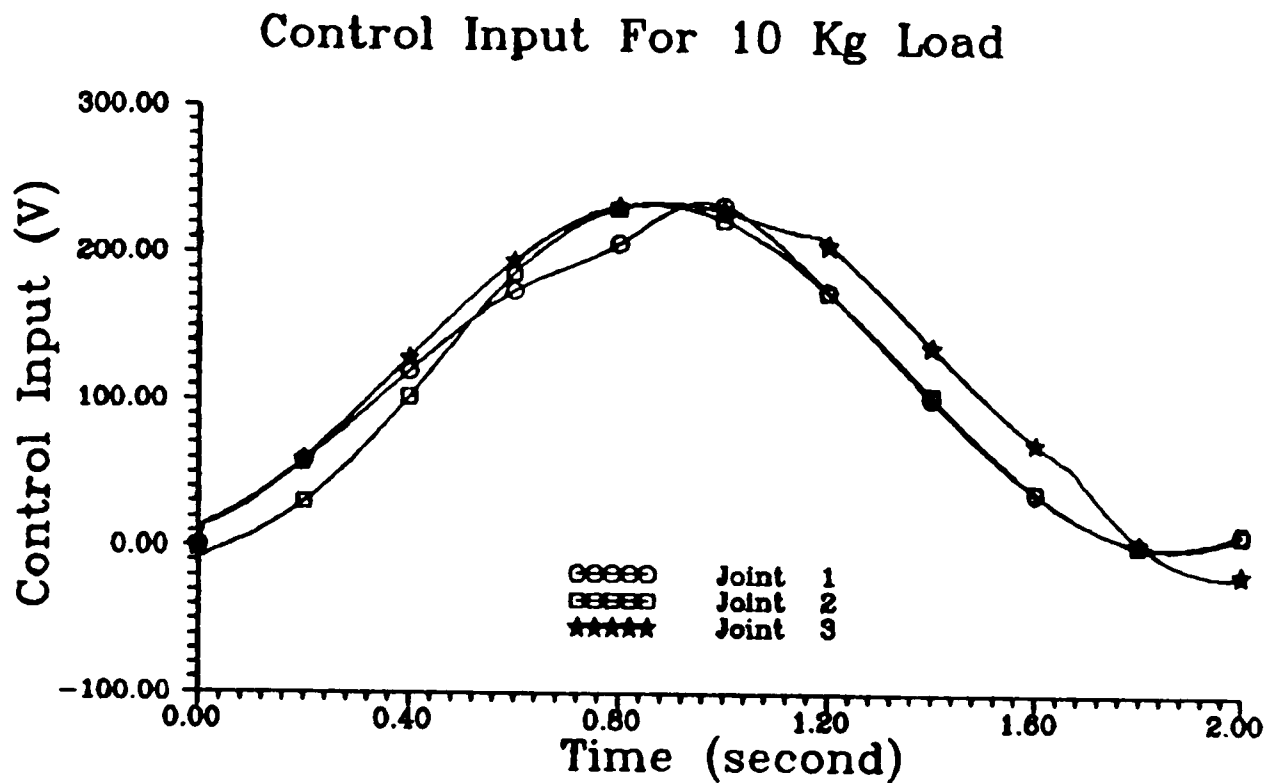


FIGURE 5.7 : Joint Control Inputs For 10 Kg. Load Using Decentralized Global Control Method

with the decentralized global control approach. This is especially pronounced for the case of joints 2 and 3 where the error has significantly decreased as compared to the decentralized local control approach.

The interaction dynamics as represented by the interconnection functions  $A_{ij}(X, \xi, t)$  affect the performance of the overall system, and their influence cannot be ignored if high tracking accuracy is desired. They have to be compensated. The results show that the proposed decentralized global control approach is capable of suppressing the effect of the interconnection functions as well as the uncertainties present in the system better than the decentralized local control algorithm.

The simulation results also support the theory that, under the same conditions, the decentralized global approach produces smaller tracking errors in comparison to the decentralized local control algorithm which do not employ the neighbouring states as part of the feedback signals (that is, do not employ any compensation for the interconnection dynamics). In other words, the uniform ultimate boundedness set for the decentralized global approach is smaller than

that of the decentralized local approach under the same conditions.

In the next simulations, the robustness of the decentralized global control approach is evaluated under different loading conditions.

#### 5.4.2 Effect Of Load Variation

Different load masses were placed at the end of the robot manipulator's joint which gave the varying effect of the manipulator linkage inertia, Coriolis, centrifugal and gravitational forces exerted on the overall robot manipulator dynamics. This will affect the effectiveness of the decentralized global control algorithm.

The manipulator was required to follow a nominal trajectory first without any load, then with a 10 Kg. load, and finally with a 20 Kg. load. The position tracking performance under the various loading conditions for joint 1, joint 2 and joint 3 are as shown in Figures 5.8, 5.9, and 5.10, respectively. Also shown in the figures (part b) are the corresponding joint tracking errors under the mentioned loading conditions. Figures 5.11 through 5.13 illustrate the joint velocity and acceleration tracking responses for joints 1 to 3, respectively.

The simulation results illustrate that the robot manipulator with the decentralized global controller is capable of tracking the desired joint position, velocity and acceleration trajectories under the various loading conditions with small tracking errors. The results shown here are much better (in terms of tracking accuracy) than those obtained using the decentralized local approach as shown in the previous chapter (Figure 4.12 through Figure 4.17).

The controller was computed based on the payload range of 0 Kg. to 20 Kg. maximum. Thus, the curves for the 0 Kg. load and the 20 Kg. load in Figures 5.8b, 5.9b and 5.10b) can be considered as representing the boundaries of the tracking error for the chosen controller parameters ( feedback gains,  $K_i$ ,  $\epsilon_i$ ,  $Q_i$ , etc.) for the three joints of the robot manipulator. For any load within the specified range carried by the manipulator along the reference trajectories, the corresponding tracking error for each joint will fall within these boundaries. The maximum difference between the 0 Kg. and the 20 Kg. load curves are at about

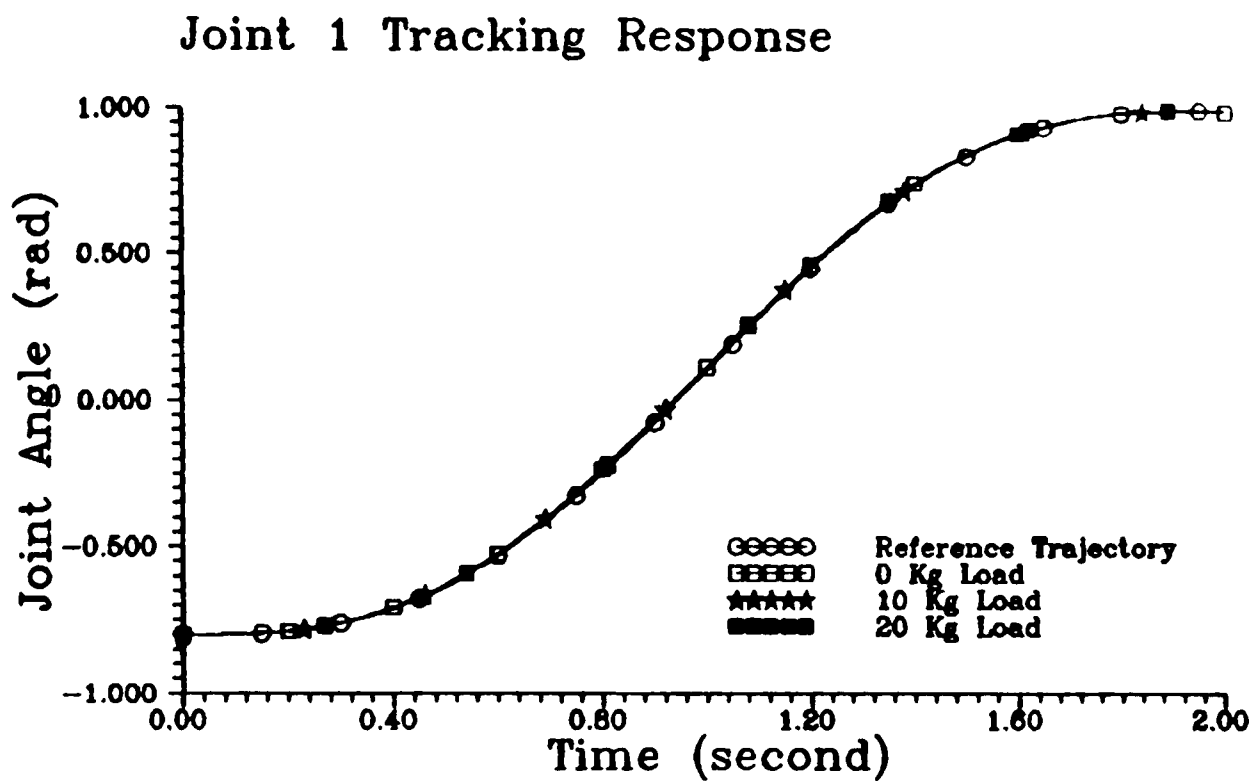


FIGURE 5.8 a : Joint 1 Tracking Response Under Different Load Using Decentralized Global Control Method

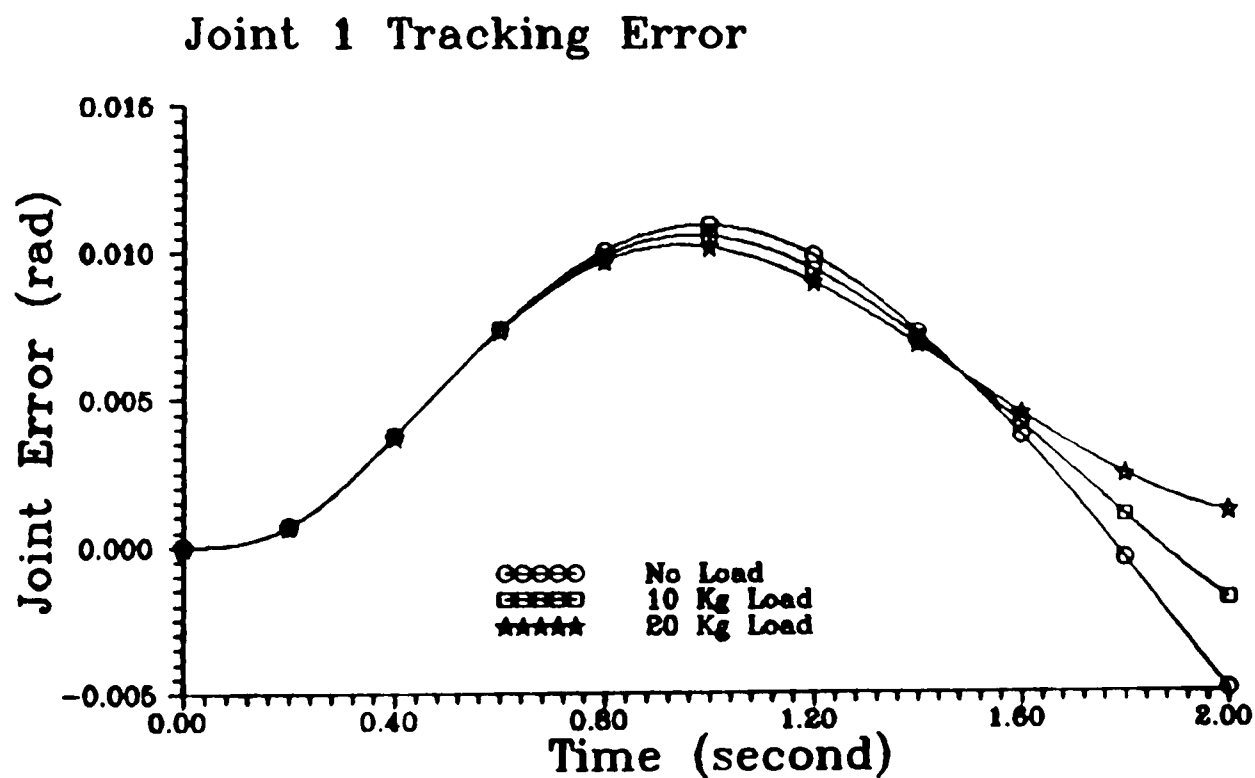


FIGURE 5.8 b : Joint 1 Tracking Error Under Different Load Using Decentralized Global Control Method

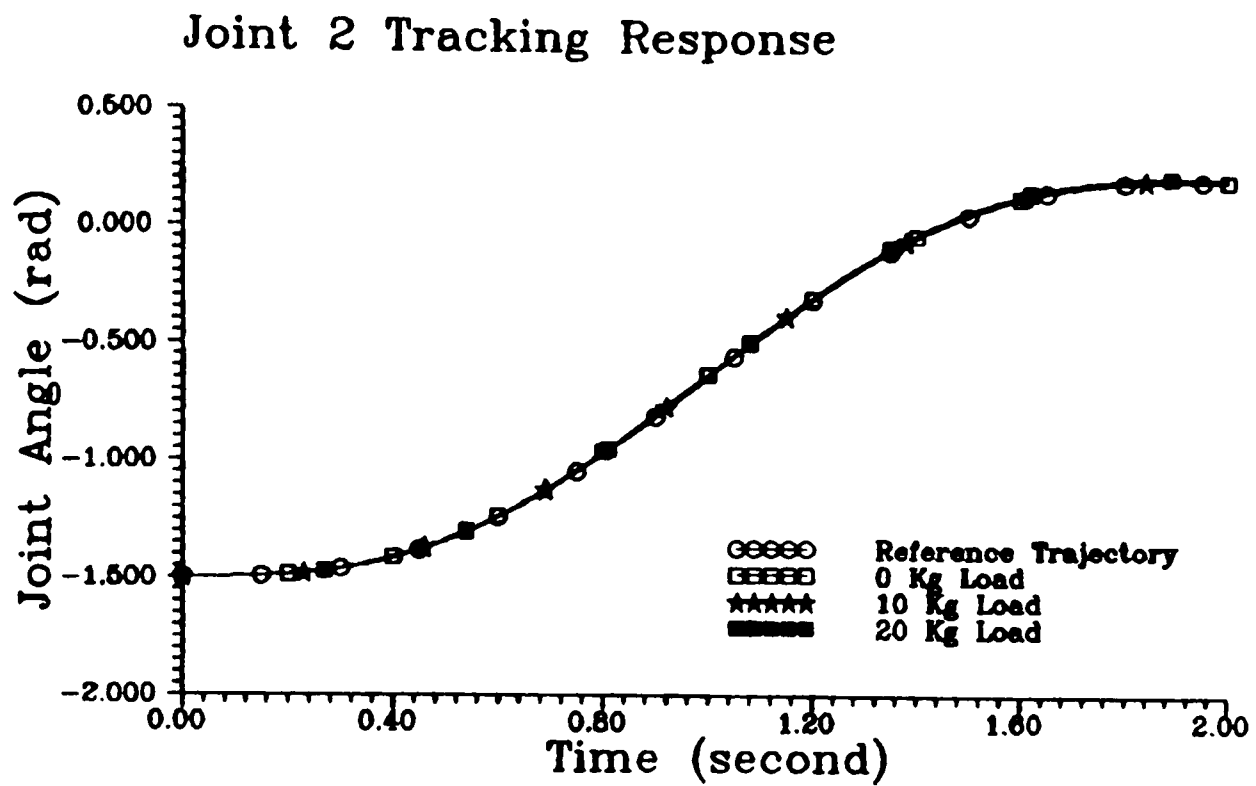


FIGURE 5.9 a : Joint 2 Tracking Response Under Different Load Using Decentralized Global Control Method

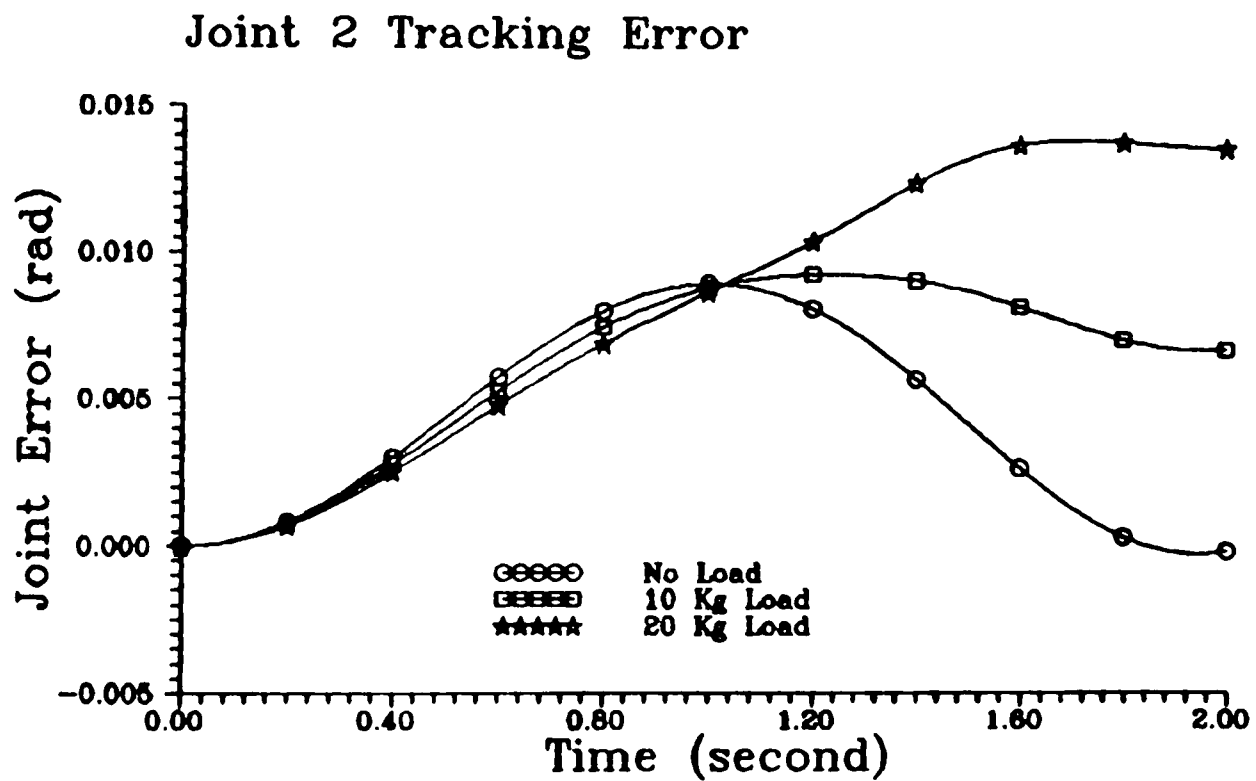


FIGURE 5.9 b : Joint 2 Tracking Error Under Different Load Using Decentralized Global Control Method

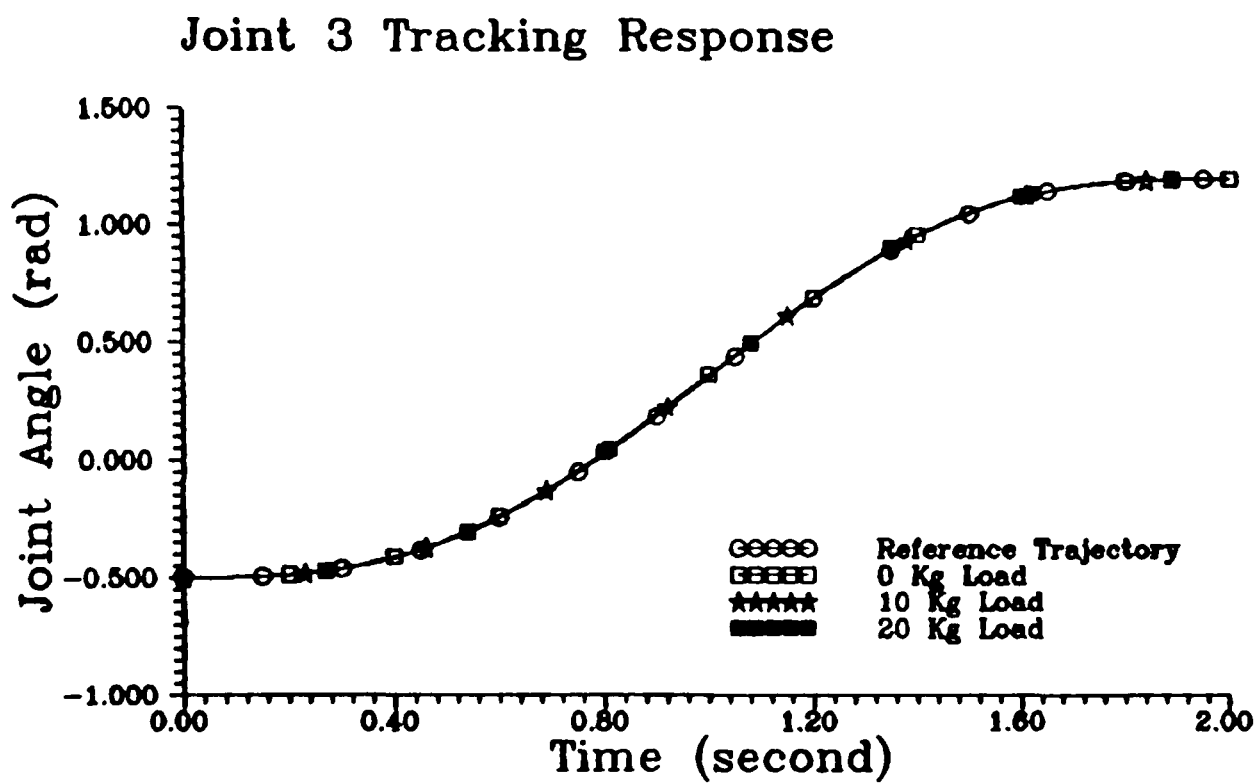


FIGURE 5.10 a : Joint 3 Tracking Response Under Different Load  
Using Decentralized Global Control Method

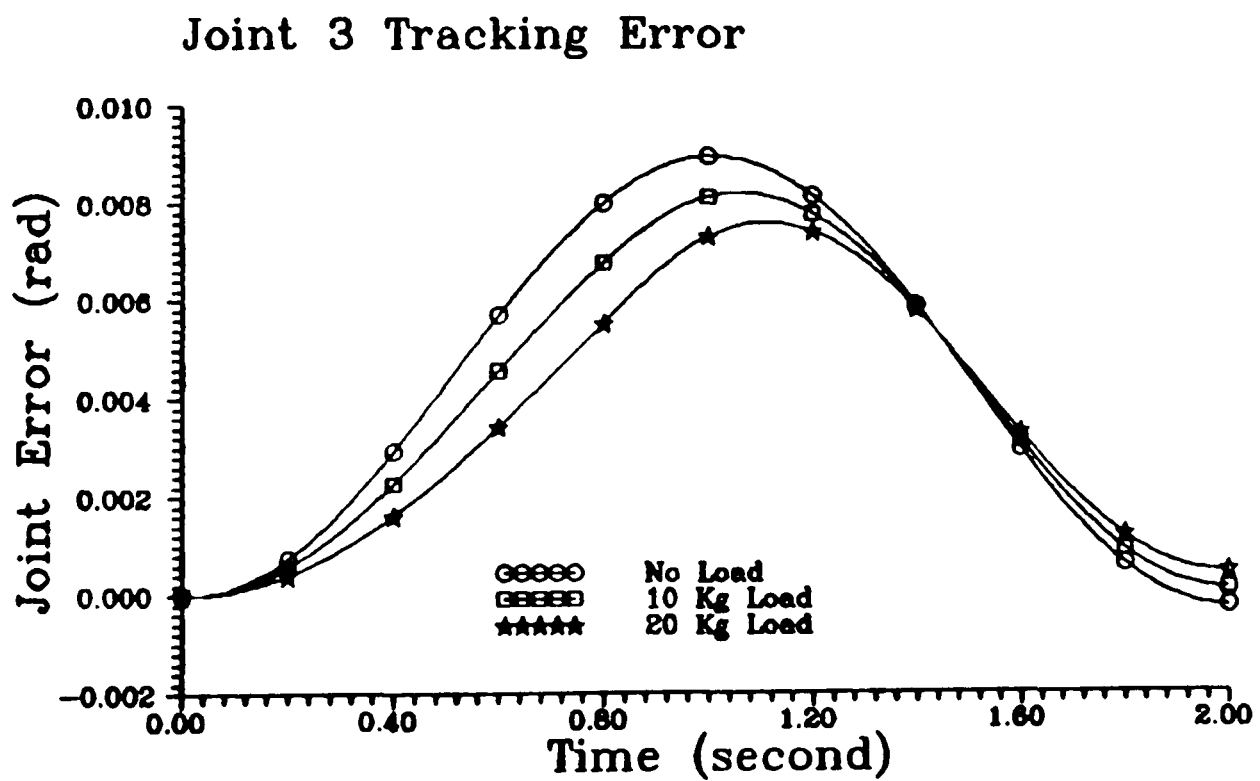


FIGURE 5.10 b : Joint 3 Tracking Error Under Different Load  
Using Decentralized Global Control Method

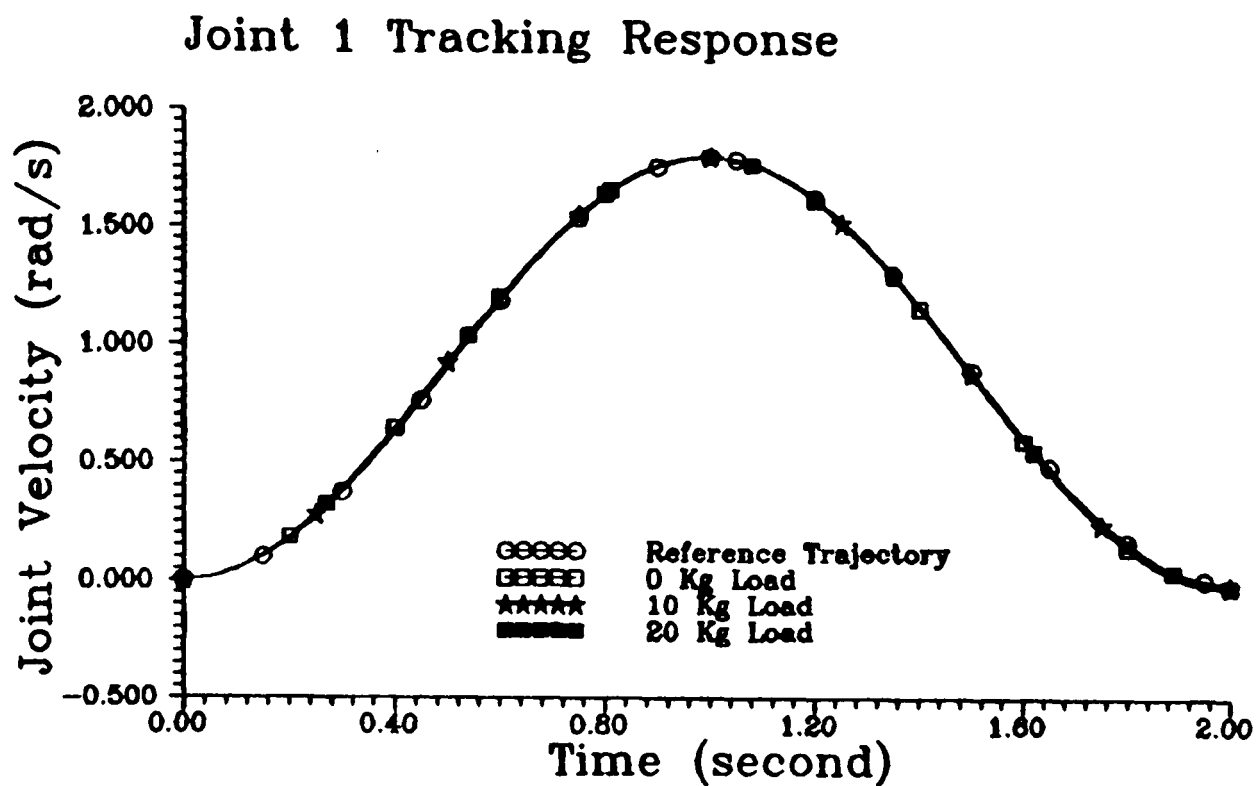


FIGURE 5.11 a : Joint 1 Velocity Response Under Different Load Using Decentralized Global Control Method

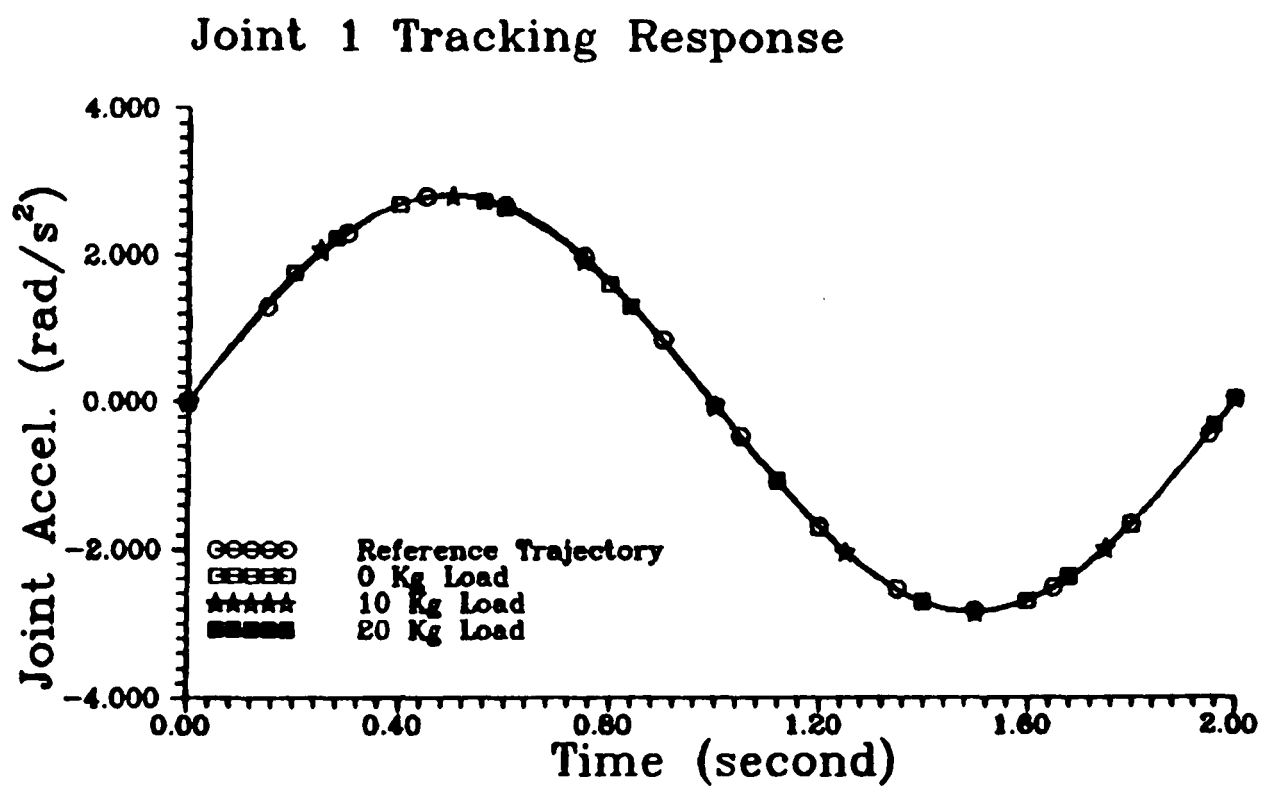


FIGURE 5.11 b : Joint 1 Acceleration Response Under Different Load Using Decentralized Global Control Method

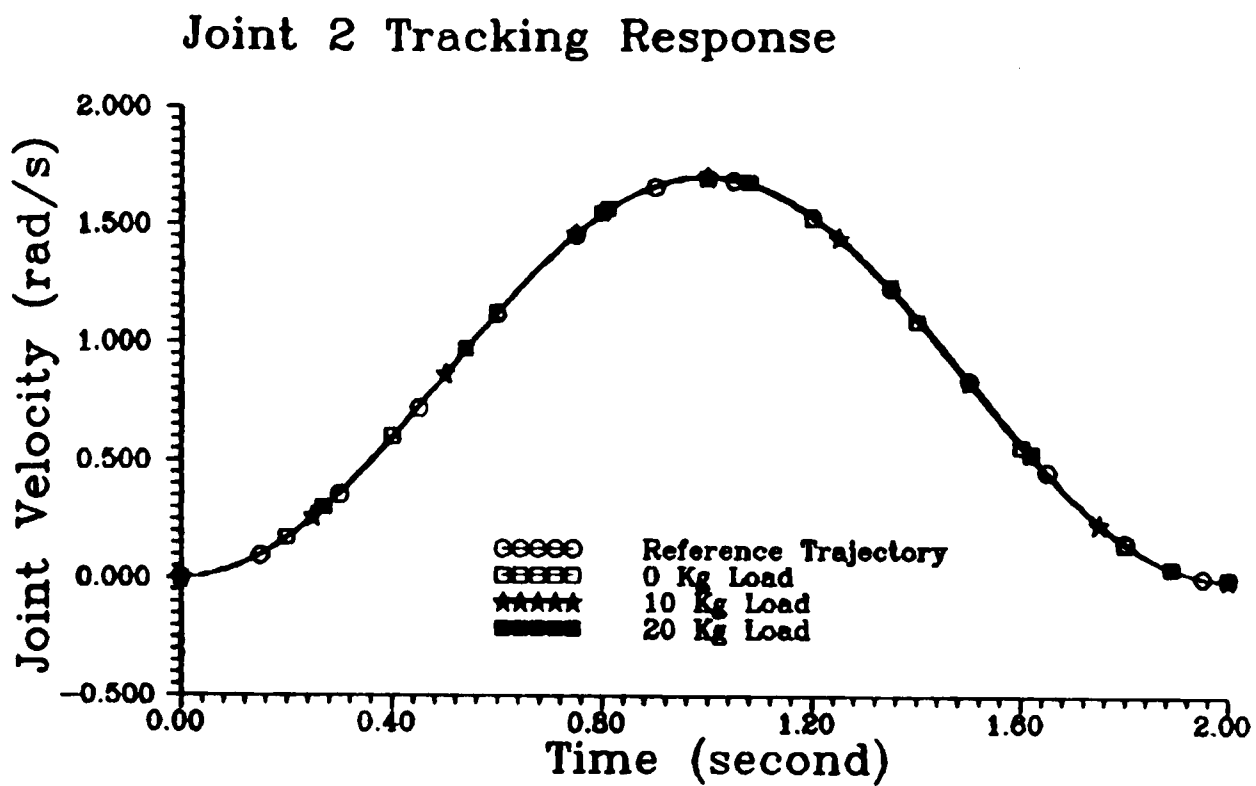


FIGURE 5.12 a : Joint 2 Velocity Response Under Different Load Using Decentralized Global Control Method

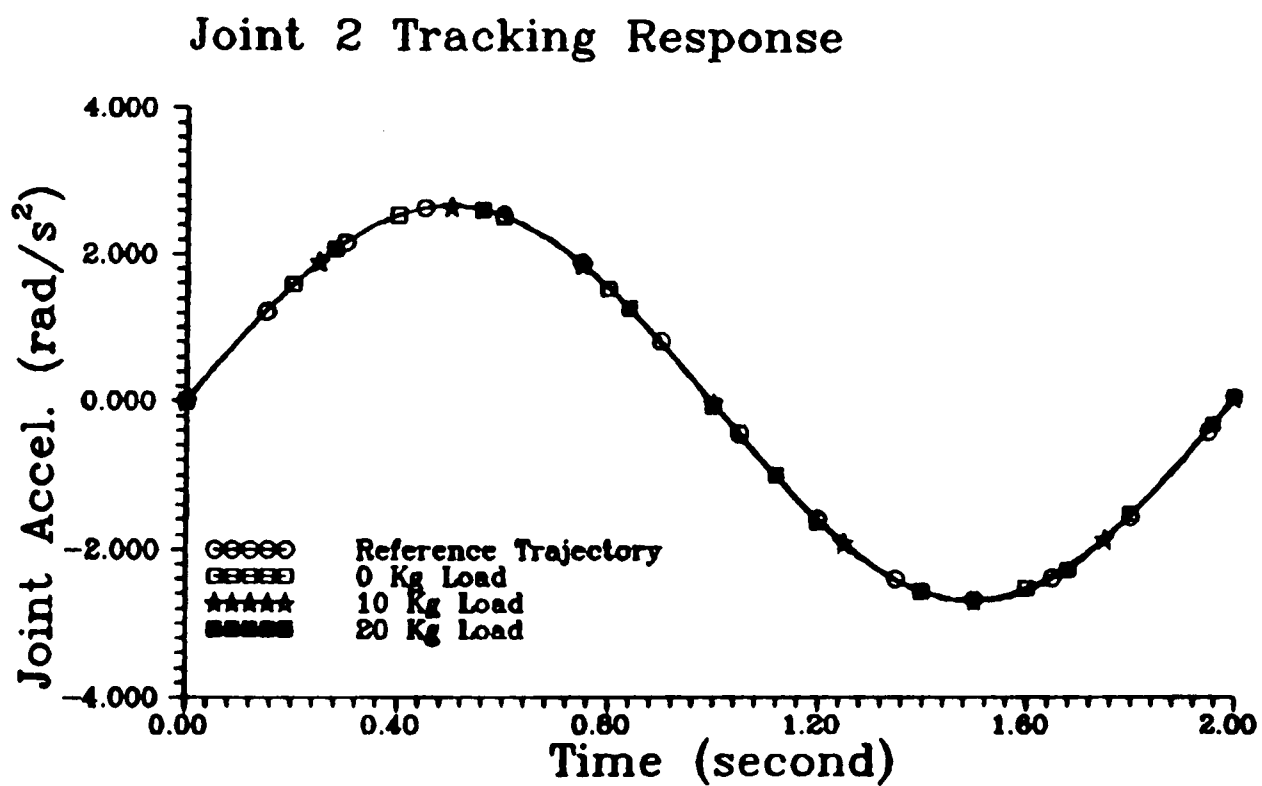


FIGURE 5.12 b : Joint 2 Acceleration Response Under Different Load Using Decentralized Global Control Method



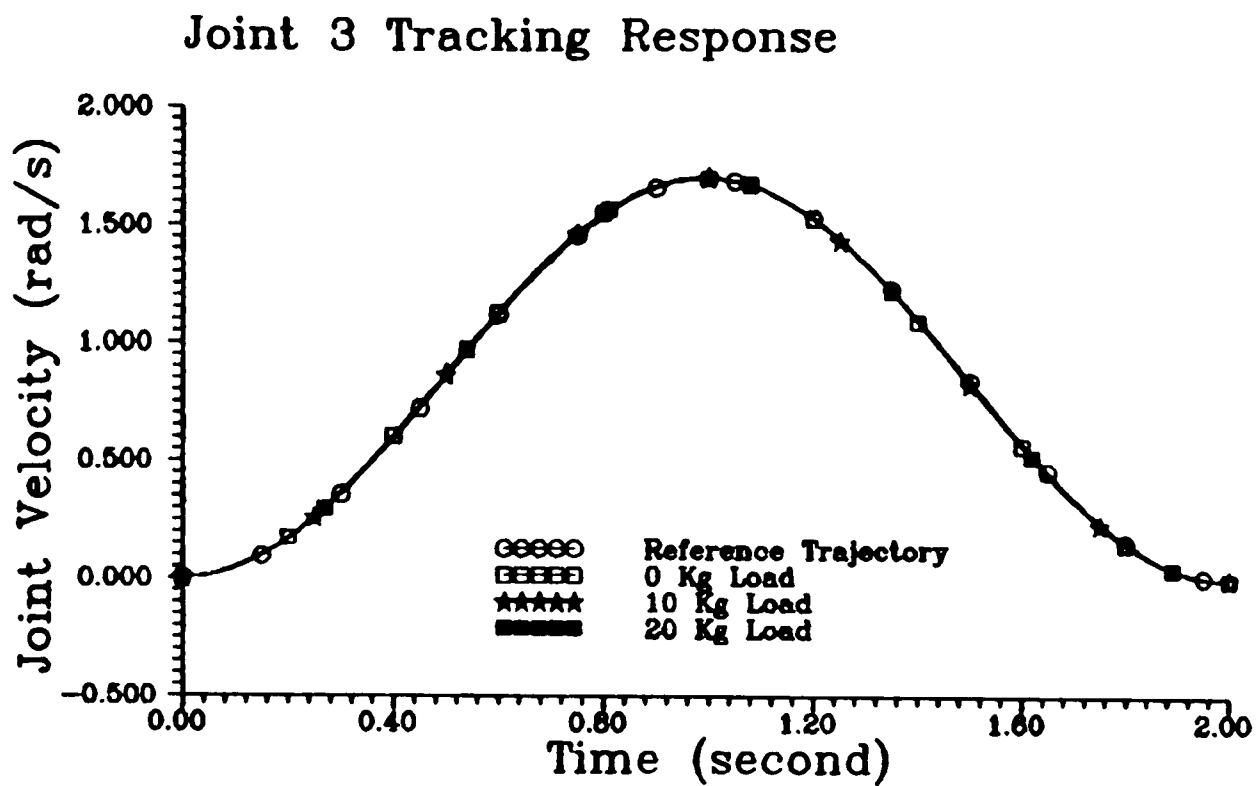


FIGURE 5.13 a : Joint 3 Velocity Response Under Different Load Using Decentralized Global Control Method

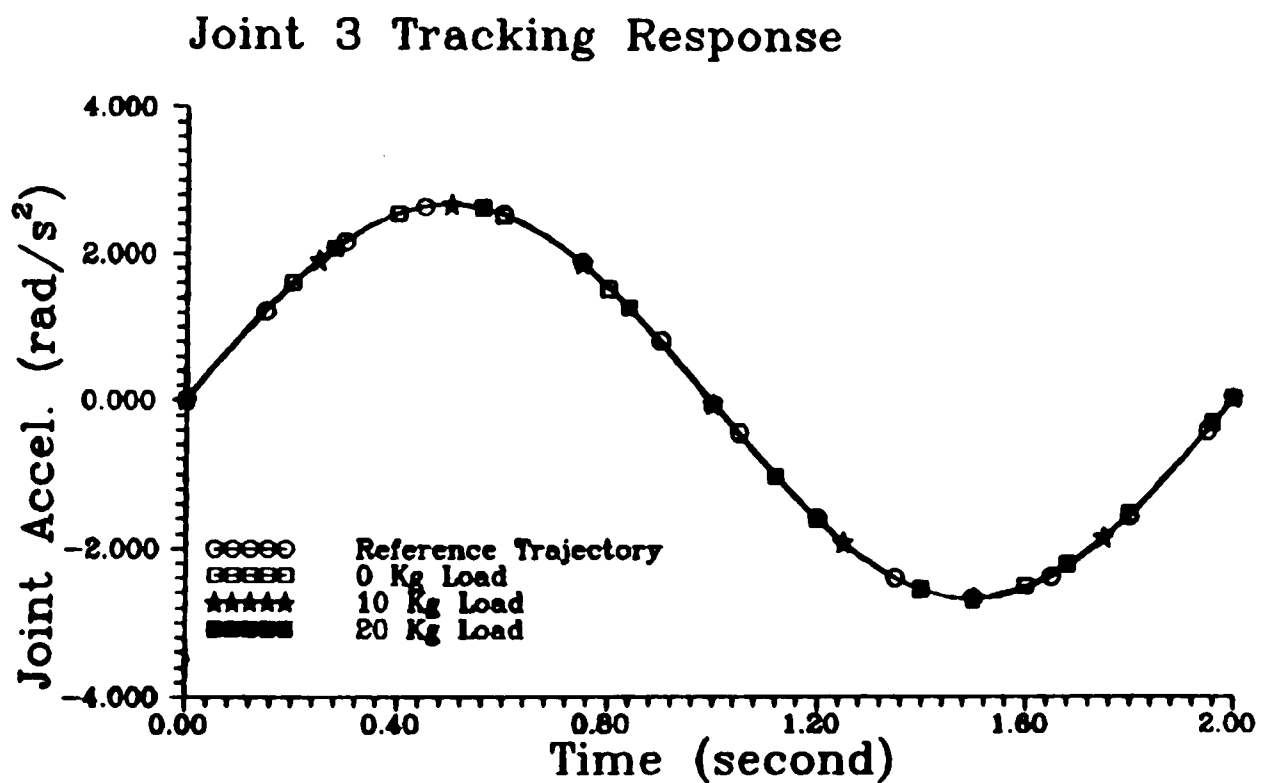


FIGURE 5.13 b : Joint 3 Acceleration Response Under Different Load Using Decentralized Global Control Method

0.006, 0.0155, and 0.0035 radians for joint 1, joint 2 and joint 3, respectively. This means that if the robot manipulator without any load can track the reference trajectories without any tracking error, the same robot manipulator with a 20 Kg. load can track the same trajectories with very small tracking errors ( with maximum error at 0.006, 0.0155, and 0.0035 radians for joint 1, joint 2, and joint 3, respectively). Furthermore, as shown in Figures 5.8a, 5.9a, and 5.10a, these tracking errors for the different loading conditions are negligible. Thus, the graphs show that high tracking accuracy can be achieved under varying loading conditions. Hence, it can be concluded that the decentralized global control law is robust to the parameter variations as the result of the different loading conditions.

Figure 5.14 and Figure 5.15 illustrate the control inputs to the joint actuators when the manipulator is moving without carrying any load and while carrying a 20 Kg. load, respectively. The control inputs are continuous, smooth, and are within the specified limits of the actuators for both cases.

The tracking accuracy may be increased further by decreasing the size of the uniform ultimate boundedness set (by reducing the radius of the closed ball enclosed by the set). There are several ways of accomplishing this. One way is by fine tuning the constant parameter  $\epsilon_i$ . The effect of varying the value  $\epsilon_i$  on the tracking accuracy of the robot manipulator utilizing the decentralized global control law is presented next.

### 5.4.3 Effect Of Varying The Value $\epsilon_i$

Varying the value  $\epsilon_i$ , in effect, varies the size of the hypothetical thin boundary layer at which the nonlinear part of the controller  $\Phi_{G_i}(Z_i, \xi, t)$  switches its form.

The manipulator is required to track a reference trajectory from an initial position of  $\theta(0) = [-0.5, -1.2, -0.2]^T$  radians to the final position of  $\theta(\tau) = [1.0, 0.2, 1.2]^T$  radians in  $\tau = 2$  seconds, while carrying a 10 Kg. load.

Four different cases have been considered. Each case corresponds to a different set of  $\epsilon_i$  values as tabulated in the following table (Table 5-1).

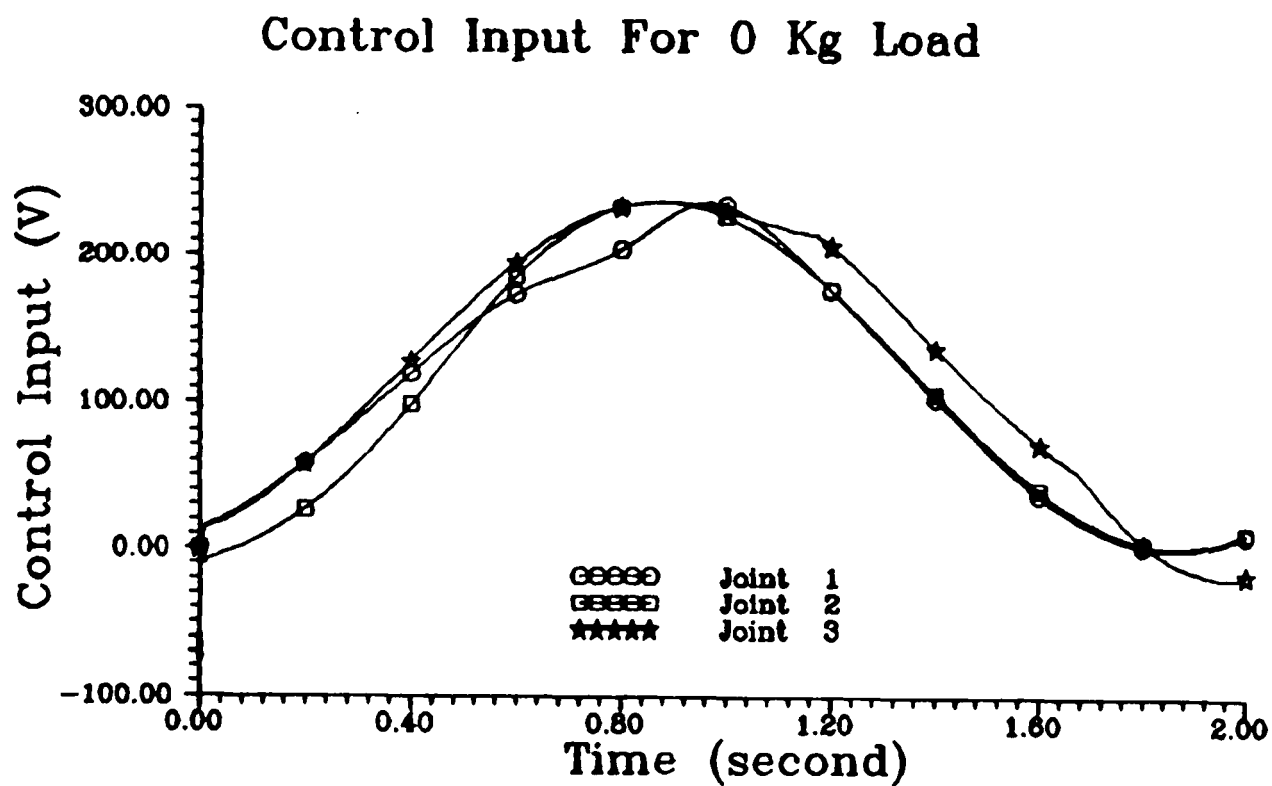


FIGURE 5.14 : Control Inputs For 0 Kg. Load Using Decentralized Global Control Method

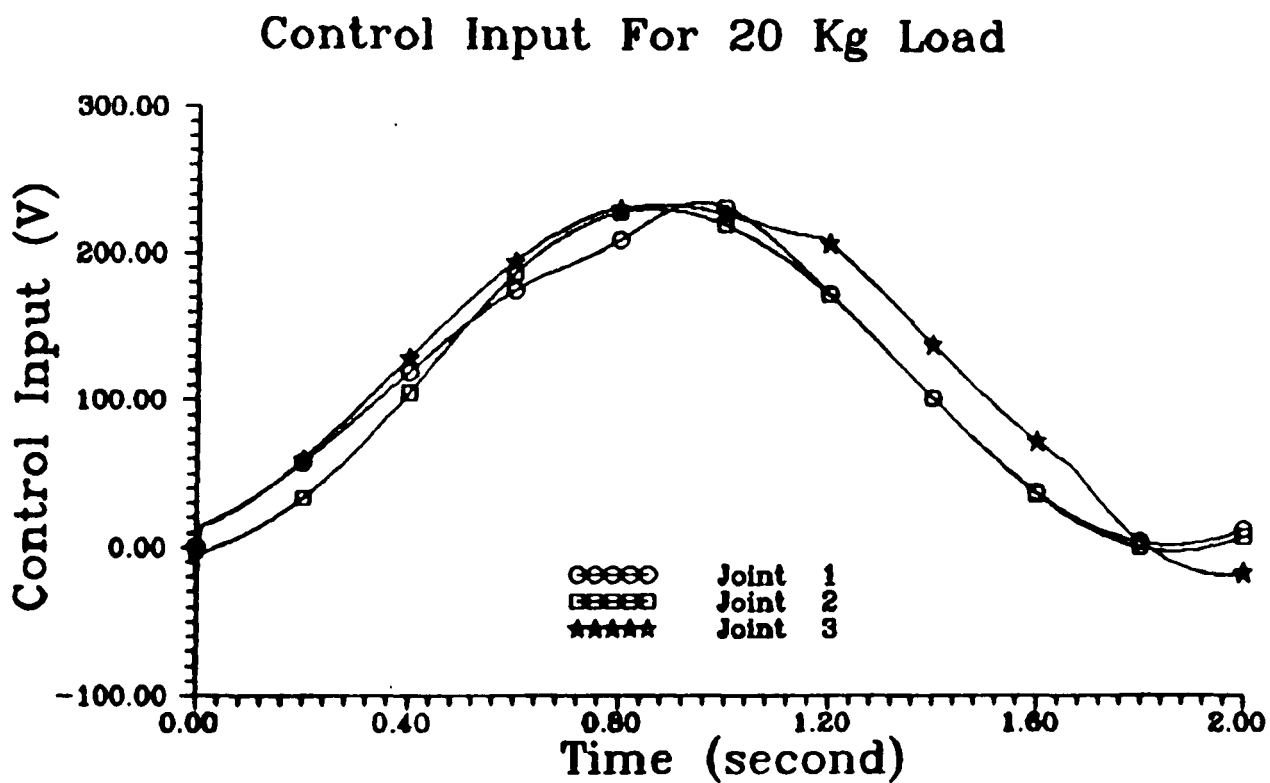


FIGURE 5.15 : Control Inputs For 20 Kg. Load Using Decentralized Global Control Method

	Subsystem			Radius Of The Closed Ball Contained In The Uniform Ultimate Boundedness Set $\eta_G = \sqrt{\frac{\sum \epsilon_i}{\lambda_{min}(\mathbb{T})}}$
	1	2	3	
	$\epsilon_1$	$\epsilon_2$	$\epsilon_3$	
Case 1	2.0	2.0	2.0	4.899
Case 2	1.5	1.5	1.5	4.243
Case 3	1.0	1.0	1.0	3.464
Case 4	0.6	0.3	0.7	2.828

**TABLE 5-1 : Different Sets Of  $\epsilon_i$  And The Corresponding  $\eta_G$**

Also listed in Table 5-1 (in the last column) are the corresponding radii of the closed ball which determined the size of the uniform ultimate boundedness set for each case. It can be seen from the table that, as the values of  $\epsilon_i$  decreases, the size of the closed ball decreases. Hence, the size of the uniform ultimate boundedness set becomes smaller. This in turn will result in tighter tracking of the desired trajectory (increase in accuracy).

The time domain simulation results for all the cases are as shown in Figure 5.16 to Figure 5.19. It can be seen from Figures 5.16, 5.17, and 5.18 that, as the values of the  $\epsilon_i$  decreases, the tracking error between the actual robot response and the desired trajectory decreases for all the joints. Hence, the tracking accuracy of the robot manipulator increases as  $\epsilon_i$  decreases.

Figure 5.19 illustrates the control input to the robot actuators for case 4. The control signals for each joint is continuous, smooth, and is within the specified operating limit of the input voltage of the actuator.

The simulation results show that the smaller the value of  $\epsilon_i$ , the tighter becomes the tracking. That is, the tracking accuracy increases as the ultimate boundedness set decreases in size. Theoretically, as the value  $\epsilon_i$  approaches zero, the tracking error will go to zero. However, for a fixed sampling period, there is a limit to which the value  $\epsilon_i$  can be reduced as illustrated in the next simulation.

In this simulation, the value  $\epsilon_i$  has been reduced to

$$\epsilon_1 = 0.4 \quad , \quad \epsilon_2 = 0.15 \quad , \quad \epsilon_3 = 0.5 \quad .$$

The radius of the closed ball for this case is at 2.05, which is smaller than that of case 4. Figure 5.20 shows the joint tracking errors while the corresponding control signals are as depicted in Figure 5.21. By comparing Figure 5.20 with Figures 5.16 to 5.18, it can be observed that there is an increase in the tracking error for each joint of the manipulator compared to the simulation results of case 4. This is due to the chattering in the control signals as shown in Figure 5.21, which degrades the performance of the robot manipulator. As explained in the previous chapter, the chattering occurs because the sampling interval used is not sufficiently small for this case. For the sampling interval considered, the chattering in the control signals will be more severe if the value  $\epsilon_i$  is reduced further. The chattering is undesirable because it may excite the unmodelled dynamic effects of the robot manipulator, or may cause rapid wear of the moving parts, and may lead to degradation in the performance of the robot manipulator. Thus, there is a trade-off between tracking accuracy and chattering of the control signal.

Therefore, in designing the nonlinear controller, the value  $\epsilon_i$  for each subsystem should be chosen as small as possible so that the smallest uniform ultimate boundedness set is obtained to ensure good tracking performance. Furthermore, care should be taken to avoid chattering in the control signals.

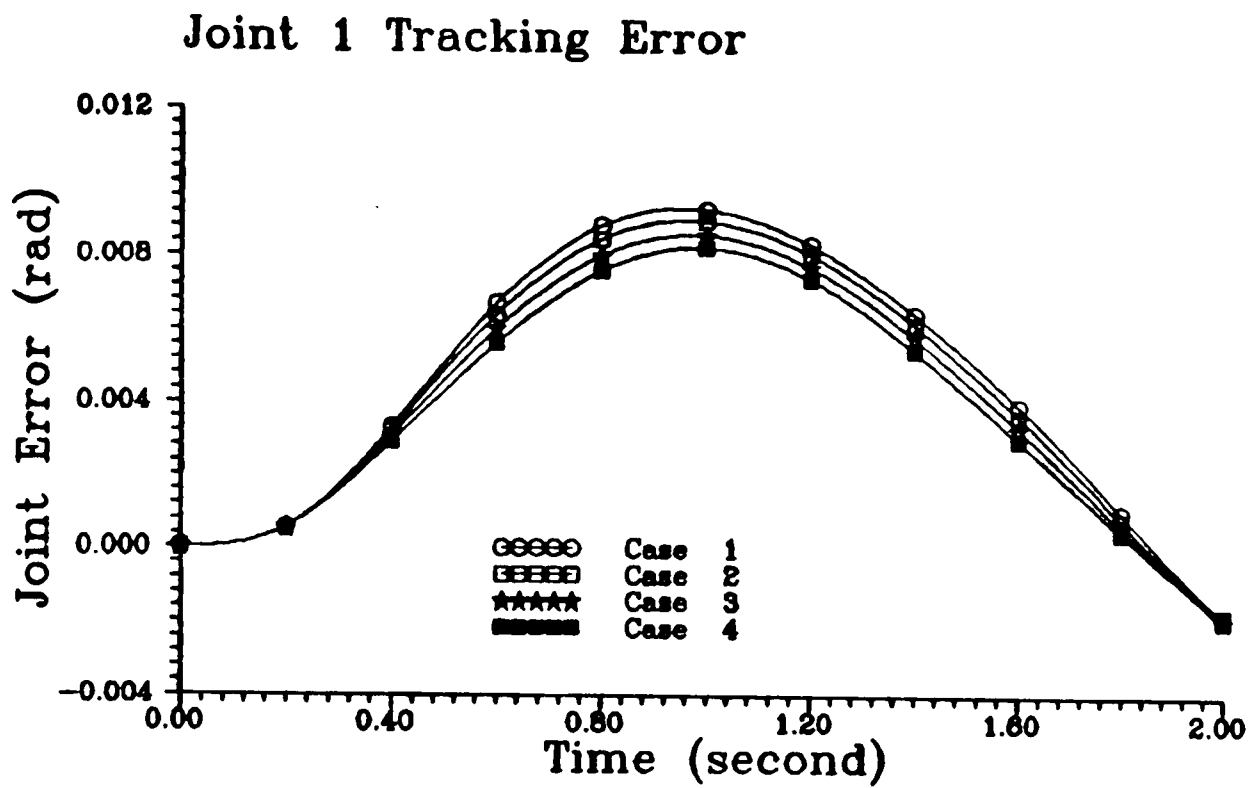


FIGURE 5.16 : Joint 1 Tracking Error For 10 Kg. Load  
For Different Values of  $\epsilon_i$

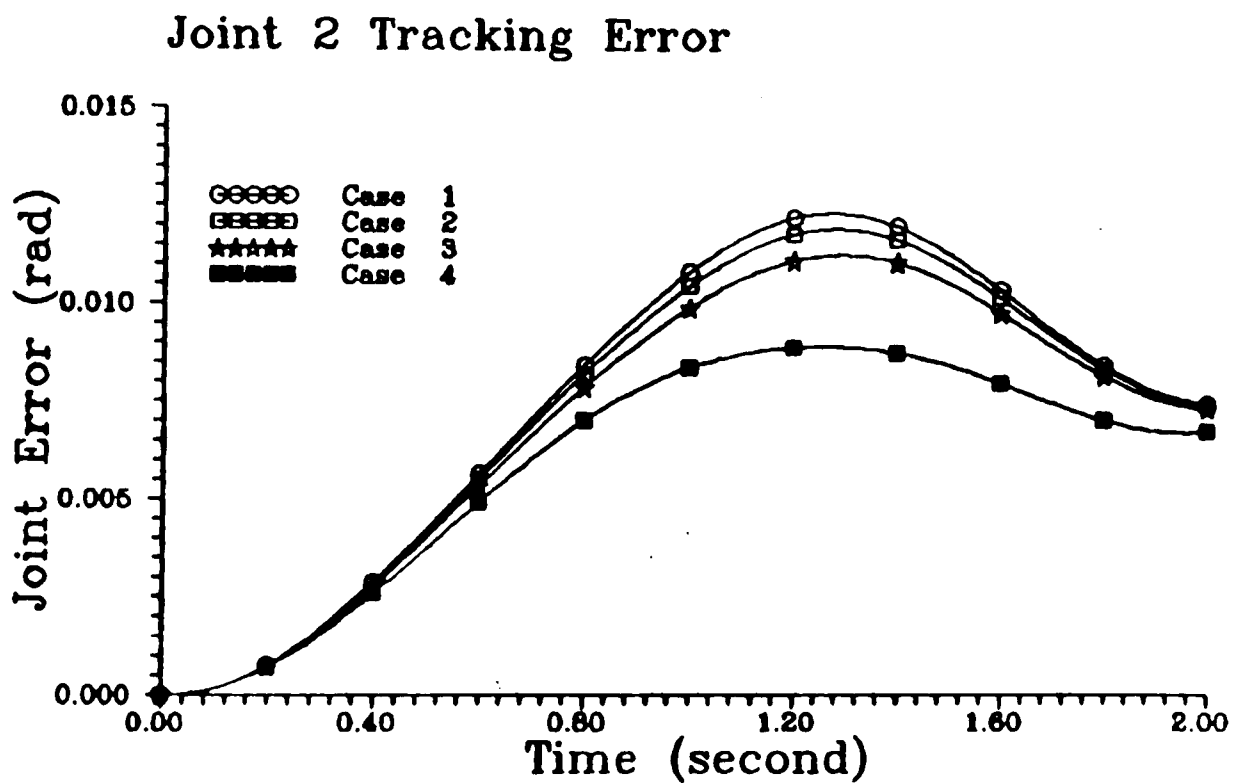


FIGURE 5.17 : Joint 2 Tracking Error For 10 Kg. Load  
For Different Values Of  $\epsilon_i$

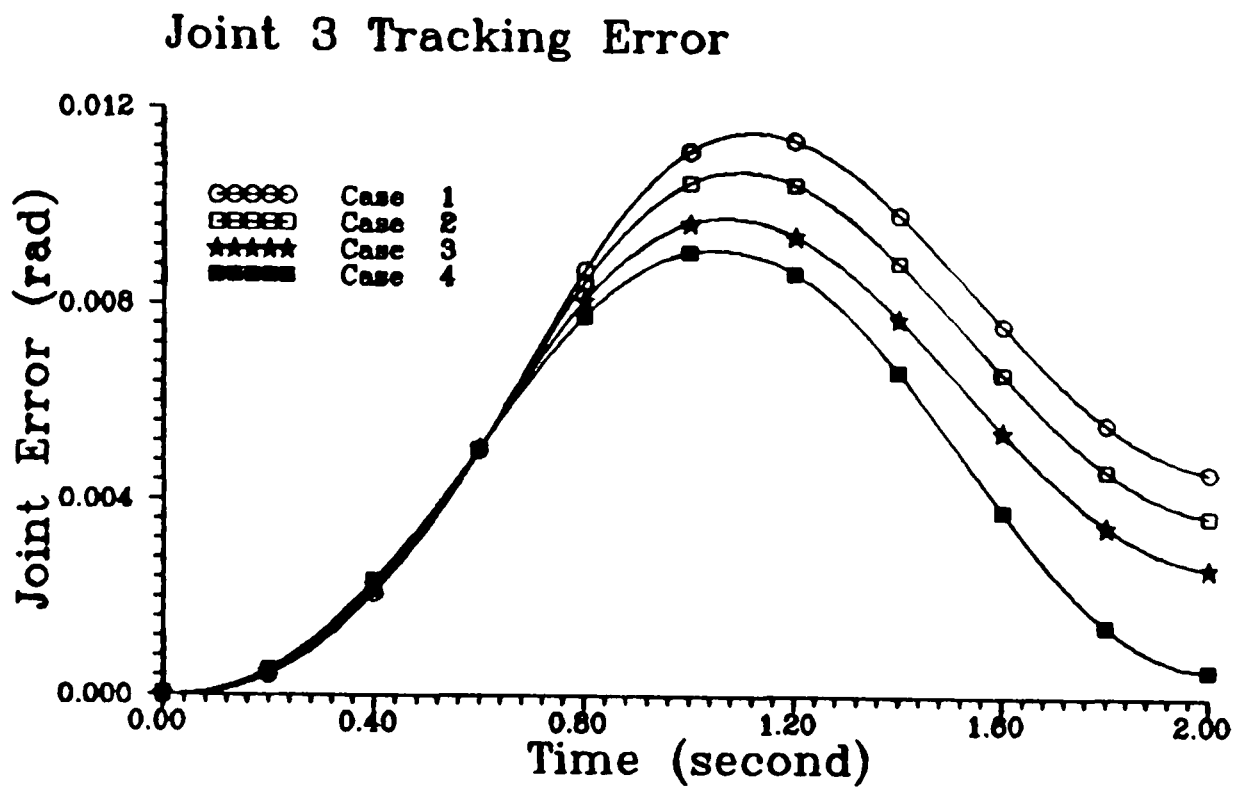


FIGURE 5.18 : Joint 3 Tracking Error For 10 Kg. Load  
For Different Values of  $\epsilon_i$

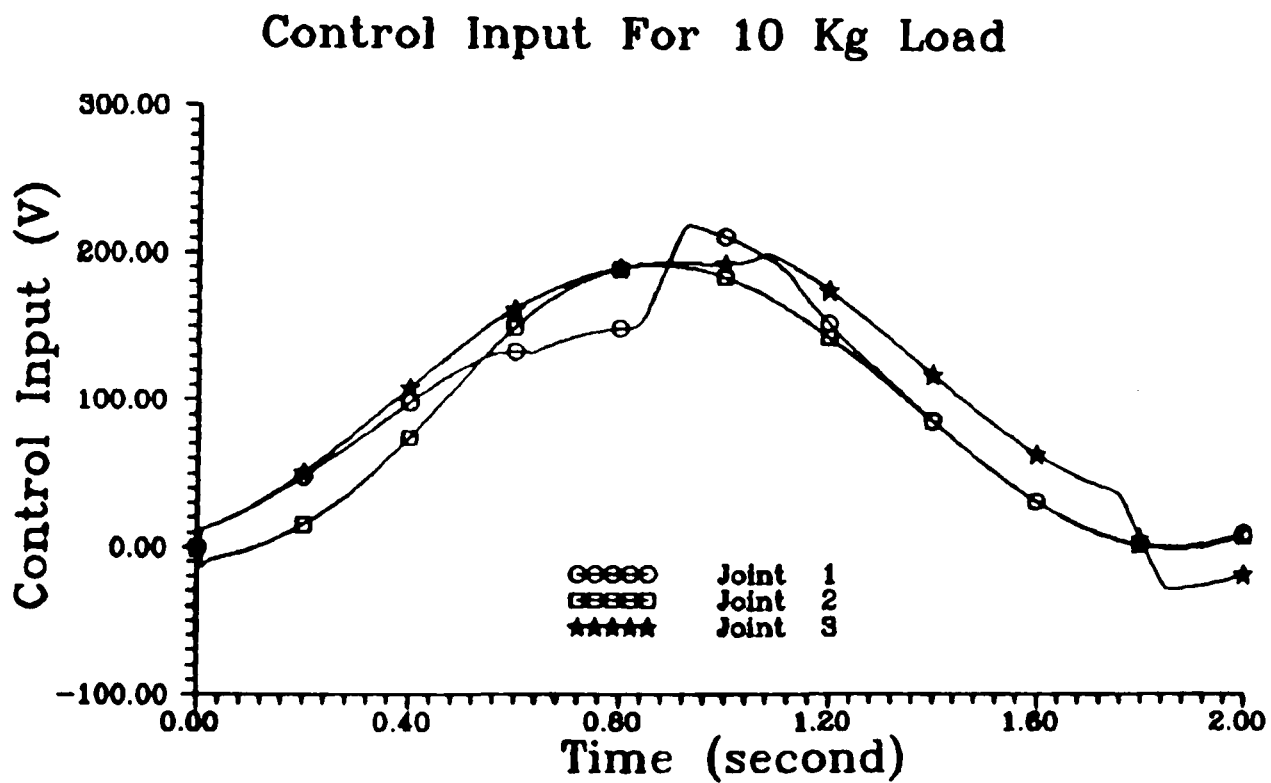


FIGURE 5.19 : Joint Control Inputs For 10 Kg. Load  
For  $\epsilon_1=0.6$  ,  $\epsilon_2=0.3$  ,  $\epsilon_3=0.7$

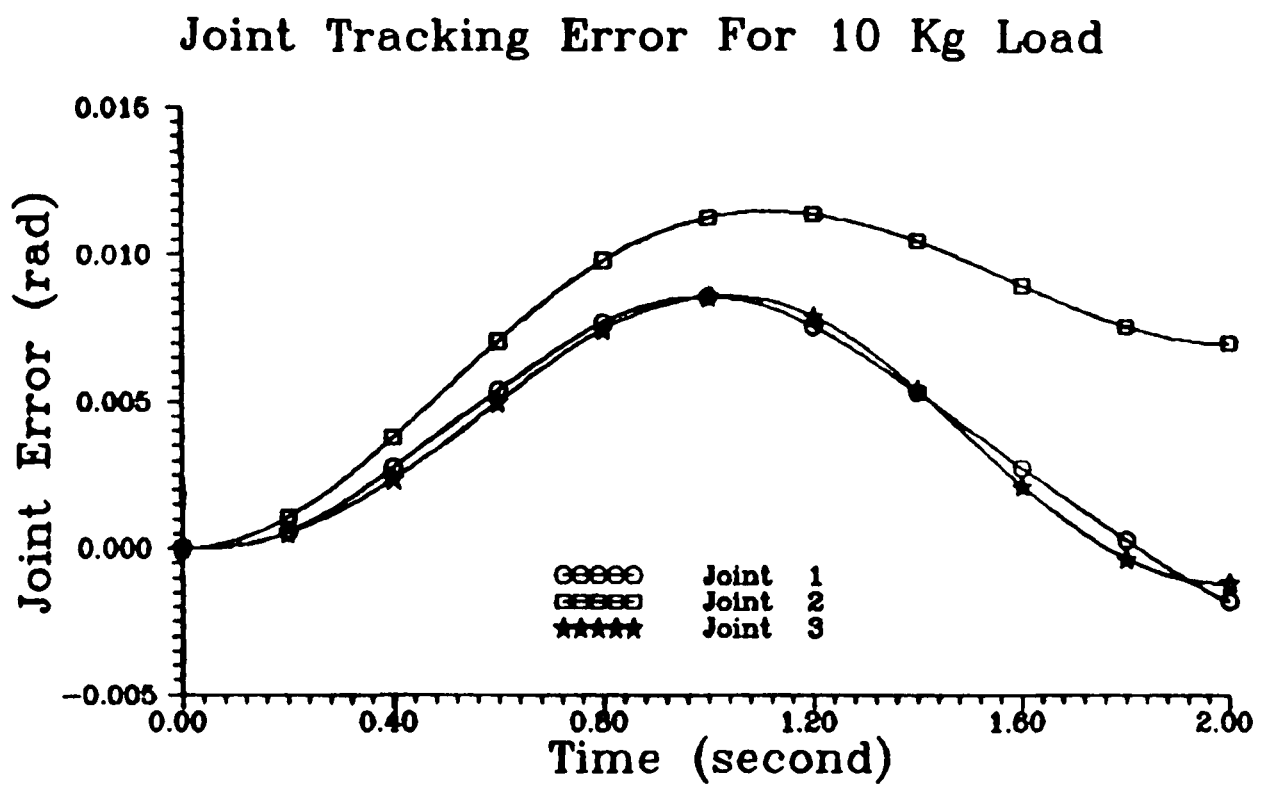


FIGURE 5.20 : Joint Tracking Errors For 10 Kg. Load  
For  $\epsilon_1=0.4$  ,  $\epsilon_2=0.15$  ,  $\epsilon_3=0.5$

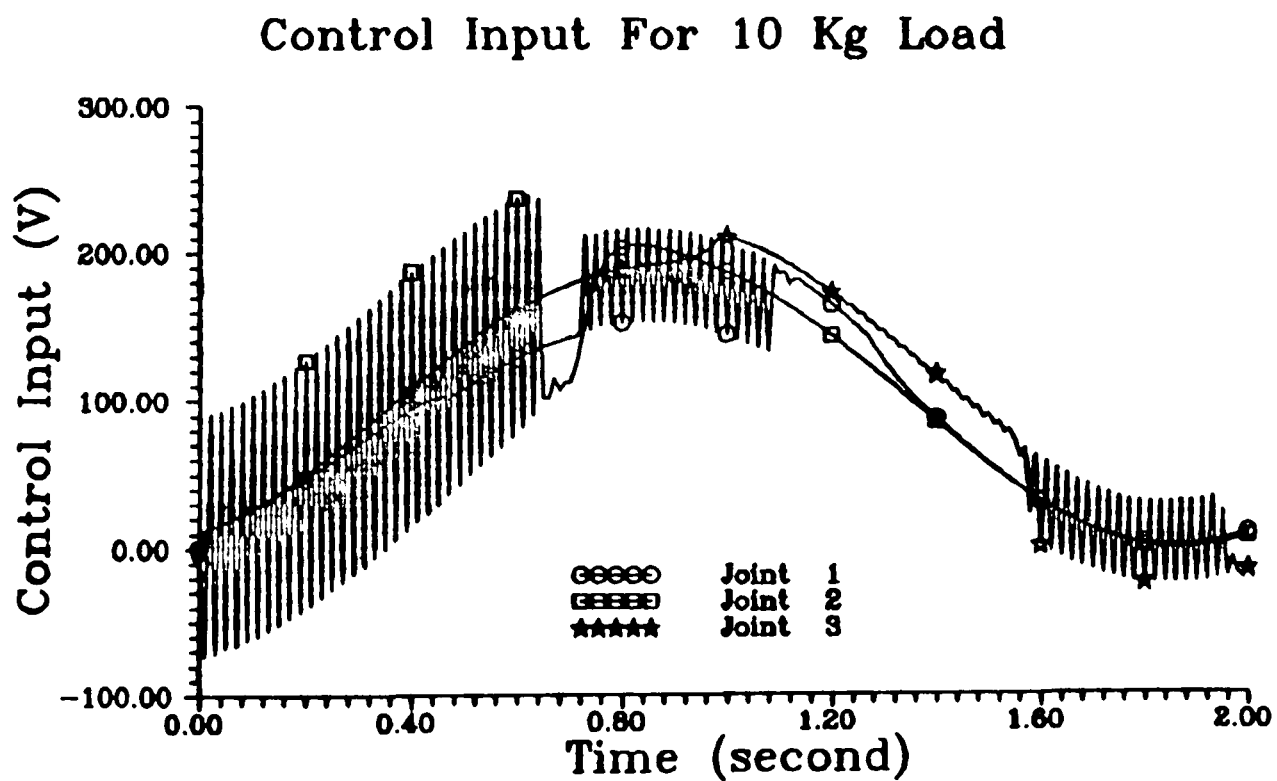


FIGURE 5.21 : Joint Control Inputs For 10 Kg. Load  
For  $\epsilon_1=0.4$  ,  $\epsilon_2=0.15$  ,  $\epsilon_3=0.5$



Apart from decreasing the value of  $\epsilon_i$ , the size of the uniform ultimate boundedness set may also be decreased by varying the weighting matrix  $Q_i$ . This will be illustrated in the following subsection.

#### 5.4.4 Effect Of Varying The Weighting Matrix $Q_i$

In this set of simulations, with  $\epsilon_i$  being kept constant, the weighting matrix  $Q_i$  was varied to evaluate the performance of the controller. The same feedback gains and tracking conditions were applied in these simulations as in the previous section. The following value of  $\epsilon_i$  has been used :

$$\epsilon_1 = 0.6 \quad , \quad \epsilon_2 = 0.3 \quad , \quad \epsilon_3 = 0.7 \quad .$$

For simplicity, it was assumed that the matrix  $Q_i$  was the same for all the three subsystems. Table 5-2 lists the different cases considered in the study.

The effect of varying  $Q_i$  on the performance of the robot manipulator may be illustrated by the joint tracking error for joint 1, joint 2, and joint3 as shown in Figures 5.22, 5.23, and 5.24, respectively, for all the cases considered. For case 1 to case 2, it can be observed from the figures that, as the matrix  $Q_i$  increases in value from, the tracking error for each joint decreases. Hence, the tracking accuracy of the robot manipulator increases as the matrix  $Q_i$  increases. The simulation results agree with the theory that as the size of the ultimate boundedness set decreases as  $Q_i$  increases. This can be observed from Table 5-2, where the radius of the closed ball wholly contained within the uniform ultimate boundedness set is decreasing as  $Q_i$  increases.

Similar to the case of  $\epsilon_i$ , there is a limit to which  $Q_i$  can be increased for fixed values of  $\epsilon_i$  and the feedback gains. Chattering will result in the control signals if  $Q_i$  is increased beyond this limit as illustrated in Figure 5.25 for case 4. This will result in a poor tracking accuracy as shown in Figures 5.22 to 5.24 for case 4.

	$Q_i$ $i = 1, 2, 3.$	Radius Of The Closed Ball Contained In The Uniform Ultimate Boundedness Set $\eta_G = \sqrt{\frac{\sum \epsilon_i}{\lambda_{\min}(\mathbb{T})}}$
Case 1	$0.1 I_3$	8.944
Case 2	$0.5 I_3$	4.0
Case 3	$I_3$	2.828
Case 4	$1.1 I_3$	2.697
	$I_3 = 3 \times 3$ Identity matrix	

TABLE 5-2 : Different Values Of  $Q_i$  And The Corresponding  $\eta_G$

Thus, in selecting an appropriate  $Q_i$  matrix for each subsystem, care should be taken to ensure good tracking performance and, at the same time, to obtain a smooth control signal that is free from chattering.

It is worthwhile at this juncture to compare the proposed decentralized method with a centralized control approach, which will be presented next.

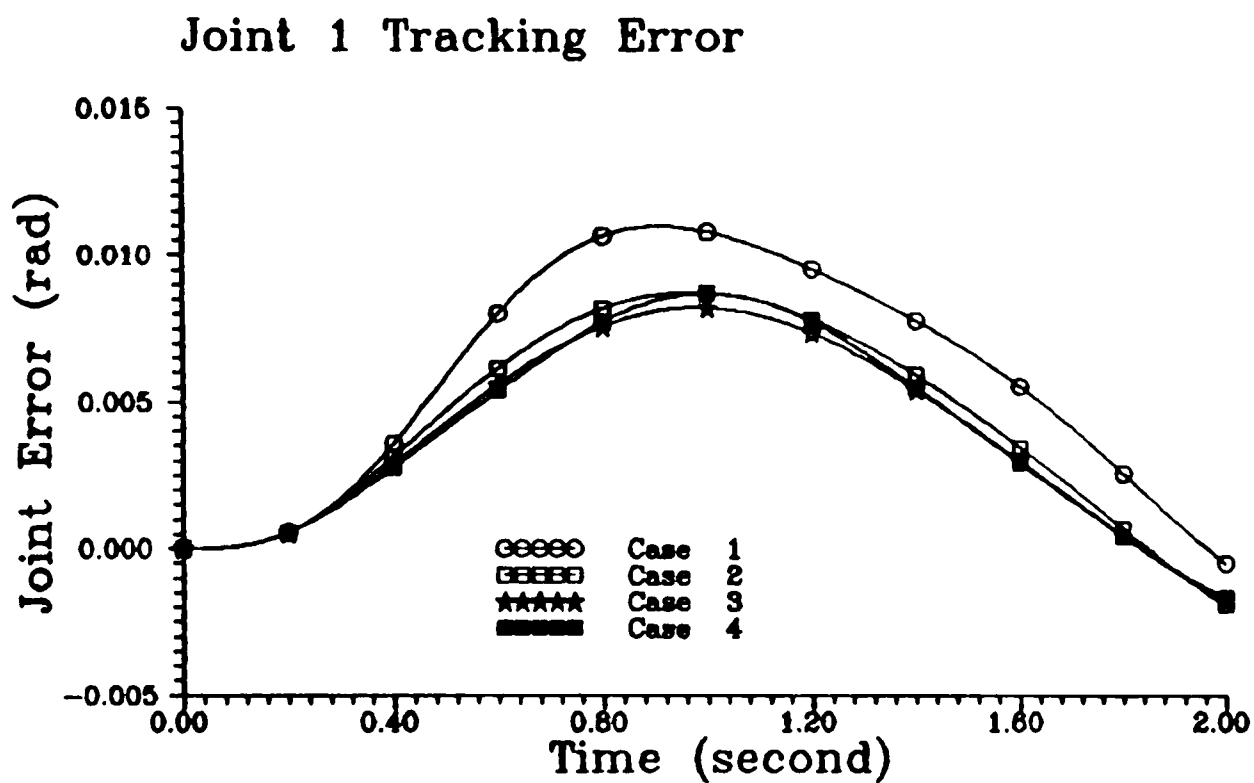


FIGURE 5.22 : Joint 1 Tracking Errors For 10 Kg. Load  
For Different  $Q_i$  Matrix

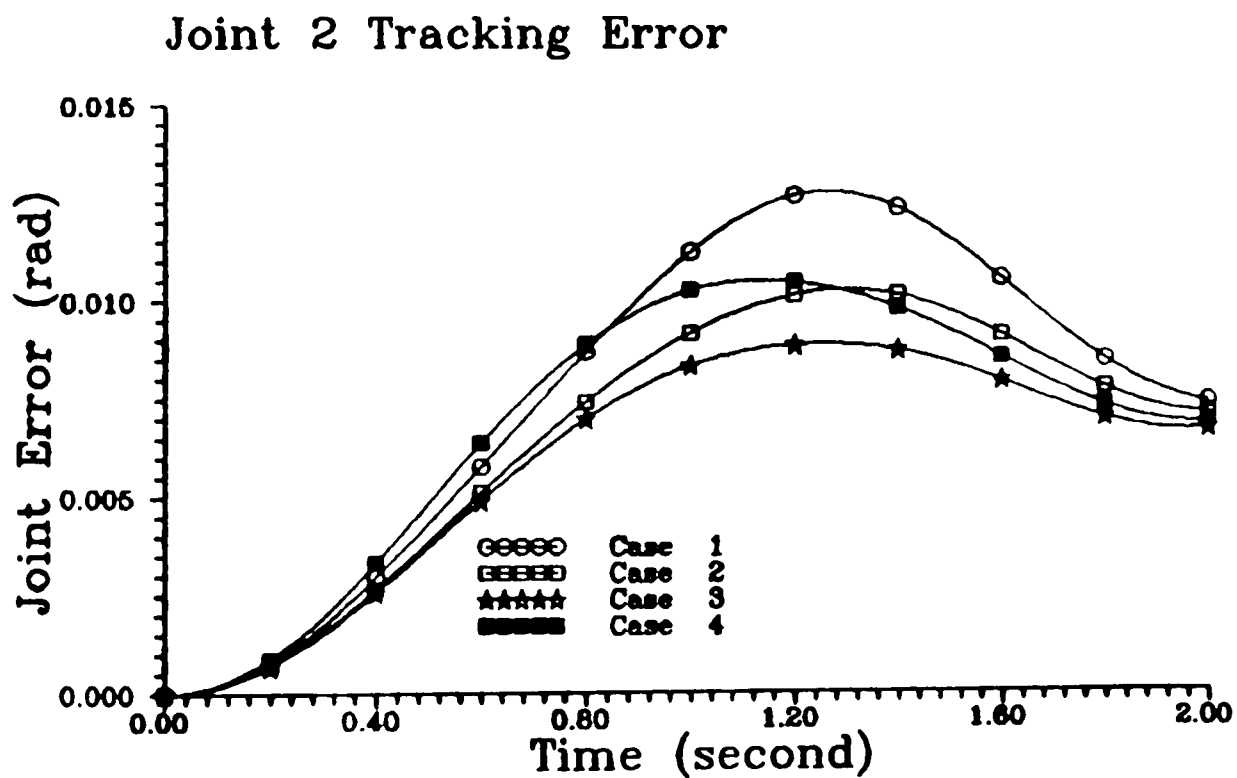


FIGURE 5.23 : Joint 2 Tracking Errors For 10 Kg. Load  
For Different  $Q_i$  Matrix

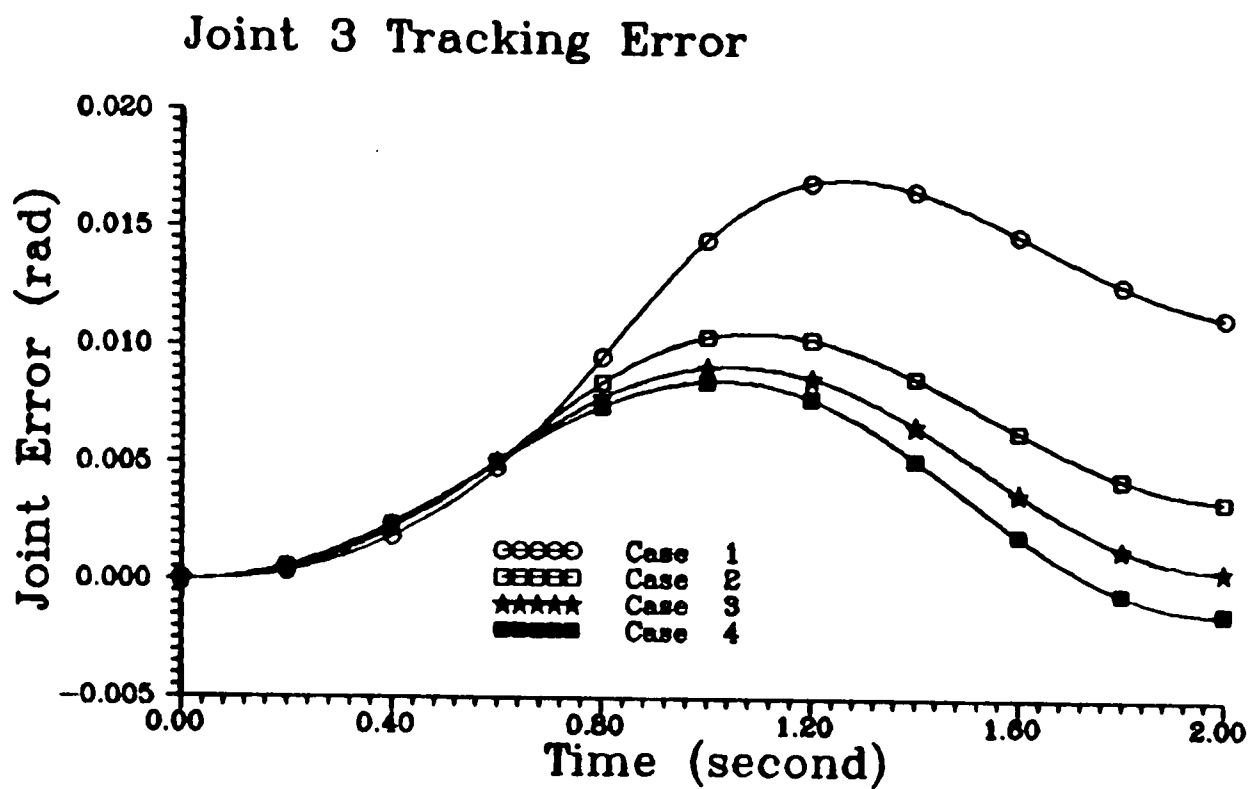


FIGURE 5.24 : Joint 3 Tracking Errors For 10 Kg. Load  
For Different  $Q_i$  Matrix

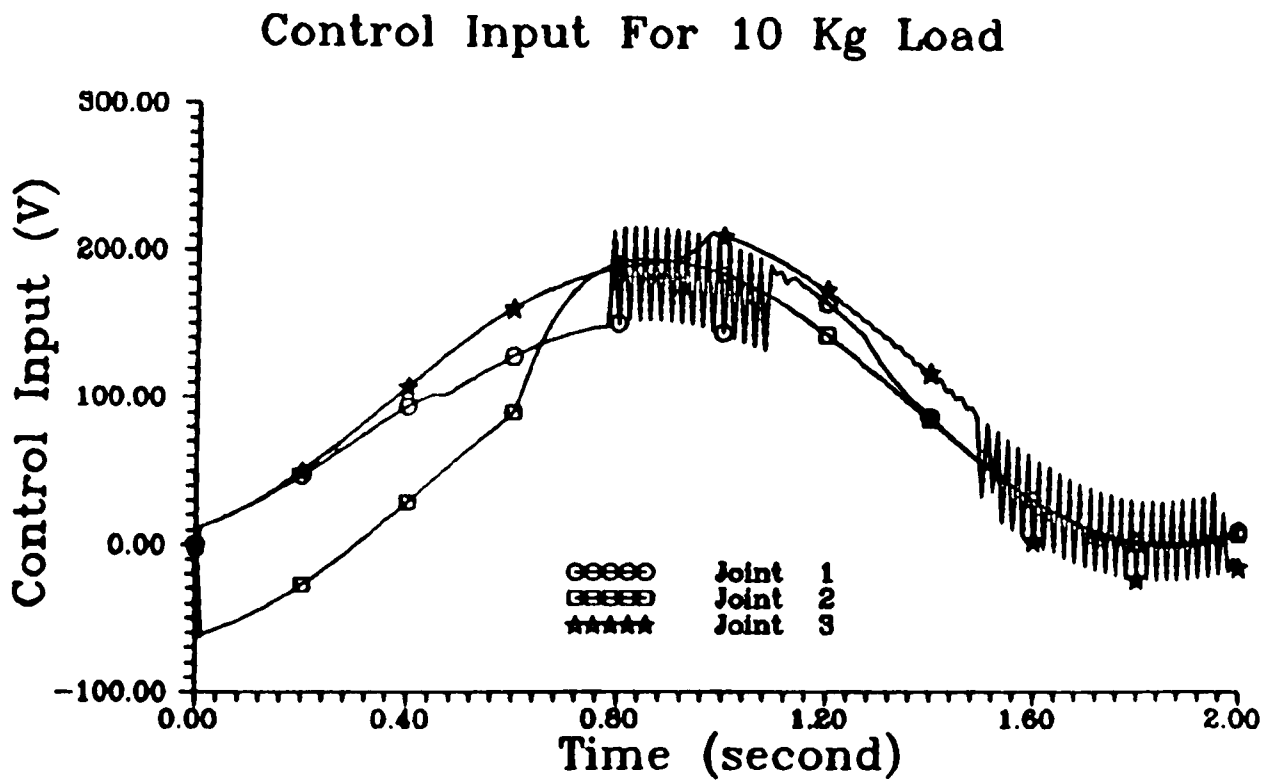


FIGURE 5.25 : Joint Control Inputs For 10 Kg.  
Load For  $Q_i = 1.1 I_3$

#### 5.4.5 Comparison Between The Decentralized and A Centralized Control Methods

The centralized control method by Corless et. al. [1984] and Shoureshi et. al. [1987; 1990] was used in the simulation study. The method is based on the deterministic approach proposed by Leitmann [1981] and thus the centralized method has the same structure as the proposed decentralized control methods. In applying the centralized control method, the overall nonlinear robot manipulator model (equation 2.76) was transformed into the following form :

$$\begin{aligned}\dot{X}(t) &= [A + \Delta A(X, \xi, t)] X(t) + [B + \Delta B(X, \xi, t)] U(t) \\ &= AX(t) + BU(t) + B [H(X, \xi, t)X(t) + E(X, \xi, t)U(t)] ,\end{aligned}\quad (5.42)$$

where

$$H(X, \xi, t) = [B^T B]^{-1} B^T \Delta A(X, \xi, t) \quad (5.43)$$

$$E(X, \xi, t) = [B^T B]^{-1} B^T \Delta B(X, \xi, t) . \quad (5.44)$$

The 3x1 centralized controller for the 3 dof robot manipulator was as follows :

$$U(t) = KZ(t) + \Omega(t) + \Phi(Z, \xi, t) . \quad (5.45)$$

where

$K$  : 3x9 feedback gain matrix

$Z(t)$  : 9x1 state error vector

$\Omega(t)$  : 3x1 vector of continuous functions

$\Phi(*)$  : 3x1 vector of nonlinear control component.

The nonlinear control component has the following form :

$$\Phi(Z, \xi, t) = \begin{cases} -\frac{\mu(Z, \xi, t)}{\|\mu(Z, \xi, t)\|} \rho(Z, \xi, t) & \text{if } \|\mu(Z, \xi, t)\| > \epsilon \\ -\frac{\mu(Z, \xi, t)}{\epsilon} \rho(Z, \xi, t) & \text{if } \|\mu(Z, \xi, t)\| \leq \epsilon \end{cases} \quad (5.46)$$

where

$$\mu(Z, \xi, t) = B^T P Z(t) \rho(Z, \xi, t) \quad (5.47)$$

$$\rho(Z, \xi, t) \triangleq \max_{\Delta b_{ij}^i \in \mathcal{Y}} [1 - \|E(\Delta b_{ij}^i)\|]^{-1} \left( \max_{\Delta a_{ij}^i \in \mathcal{R}} \|H(\Delta a_{ij}^i) X(t)\| + \max_{\Delta b_{ij}^i \in \mathcal{Y}} \|E(\Delta b_{ij}^i) [KZ(t) + \Omega(t)]\| \right). \quad (5.48)$$

The 3x1 vector  $\Omega(t)$  has been computed as follows :

$$\Omega(t) = \begin{bmatrix} 1.4313 & 0.0 & 0.0 \\ 0.0 & 1.6006 & 0.0161 \\ 0.0 & 0.0161 & 1.5752 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1(t) \\ \ddot{\theta}_2(t) \\ \ddot{\theta}_3(t) \end{bmatrix} + \begin{bmatrix} 0.0 & 125.6744 & 15.6788 \\ 0.0 & -7.357 \times 10^{-4} & 1.752 \times 10^{-5} \\ 0.0 & 0.0182 & 1.714 \times 10^{-3} \end{bmatrix} \begin{bmatrix} 0.0 & 0.03676 & 5.959 \times 10^{-17} \\ 0.0 & 141.0512 & 17.5329 \\ 0.0 & 1.852 \times 10^{-3} & 0.1664 \end{bmatrix} \begin{bmatrix} 0.0 & -0.0174 & -1.283 \times 10^{-3} \\ 0.0 & 0.0134 & 0.1405 \\ 0.0 & 141.3783 & 17.2908 \end{bmatrix} X_d(t). \quad (5.49)$$

For this simulation, the following feedback gain values were used (the same values were used for the decentralized case for comparison purposes) :

$$K = \begin{bmatrix} -3.5781 & 111.004 & -0.065 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & -400.0979 & -58.9978 & -14.4766 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ & & & 0.0 & 0.0 & 0.0 \\ & & & 0.0 & 0.0 & 0.0 \\ & & & -2.5201 & 128.5255 & 0.2785 \end{bmatrix}, \quad (5.50)$$

and  $\epsilon = 1.0$  .

In solving the matrix Lyapunov equation  $\tilde{P} \bar{A} + \bar{A}^T \tilde{P} = -Q$ , where  $\bar{A} \triangleq A+BK$  is the overall closed loop system matrix, the positive definite matrix  $Q$  has been chosen as a 9x9 identity matrix.

For the decentralized controller, the control law as designed in section 5.4.1 has been used with the above feedback gains (equations 5.50) with  $\epsilon_1 = 0.5$  ,  $\epsilon_2 = 0.3$  ,  $\epsilon_3 = 0.5$  .

Figures 5.26 to 5.28 illustrate the tracking errors of the robot manipulator joints under the decentralized global control law and the centralized control method. For comparison purposes, also shown in the figures are the tracking errors for the robot manipulator joints utilizing the decentralized local controller and the linear independent joint control approach as discussed in the previous chapter. In all the cases, the manipulator was required to track a reference trajectory from an initial position of  $\theta(0) = [ -0.8 , -1.5 , -0.5 ]^T$  radians to the final position of  $\theta(\tau) = [ 1.0 , 0.2 , 1.2 ]^T$  radians in  $\tau = 2$  seconds while carrying a 10Kg. load.

Among the four control methods, the simulation results show that the performance of the control methods based on the deterministic approach which are capable of compensating the uncertainties and nonlinearities present in the robot manipulator system are far better than the time invariant linear indepen-

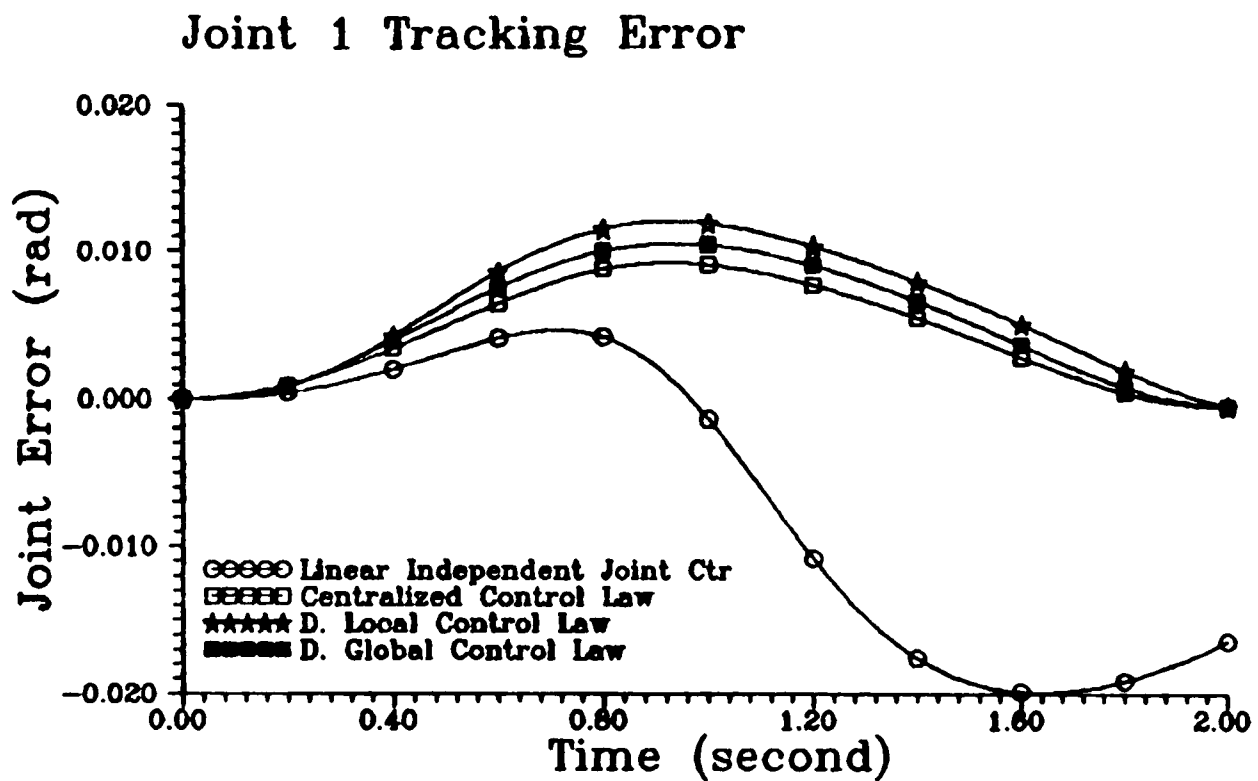


FIGURE 5.26 : Joint 1 Tracking Response Using Different Control Methods

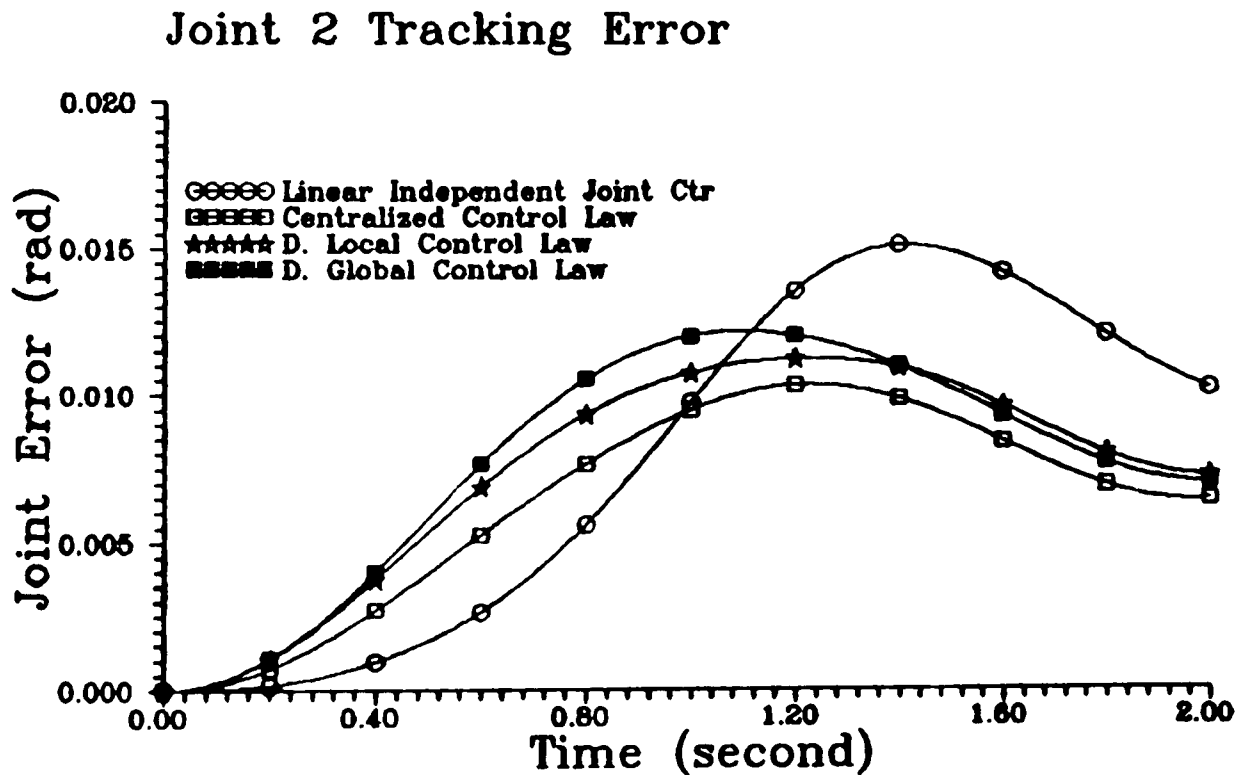


FIGURE 5.27 : Joint 2 Tracking Response Using Different Control Methods



### Joint 3 Tracking Error

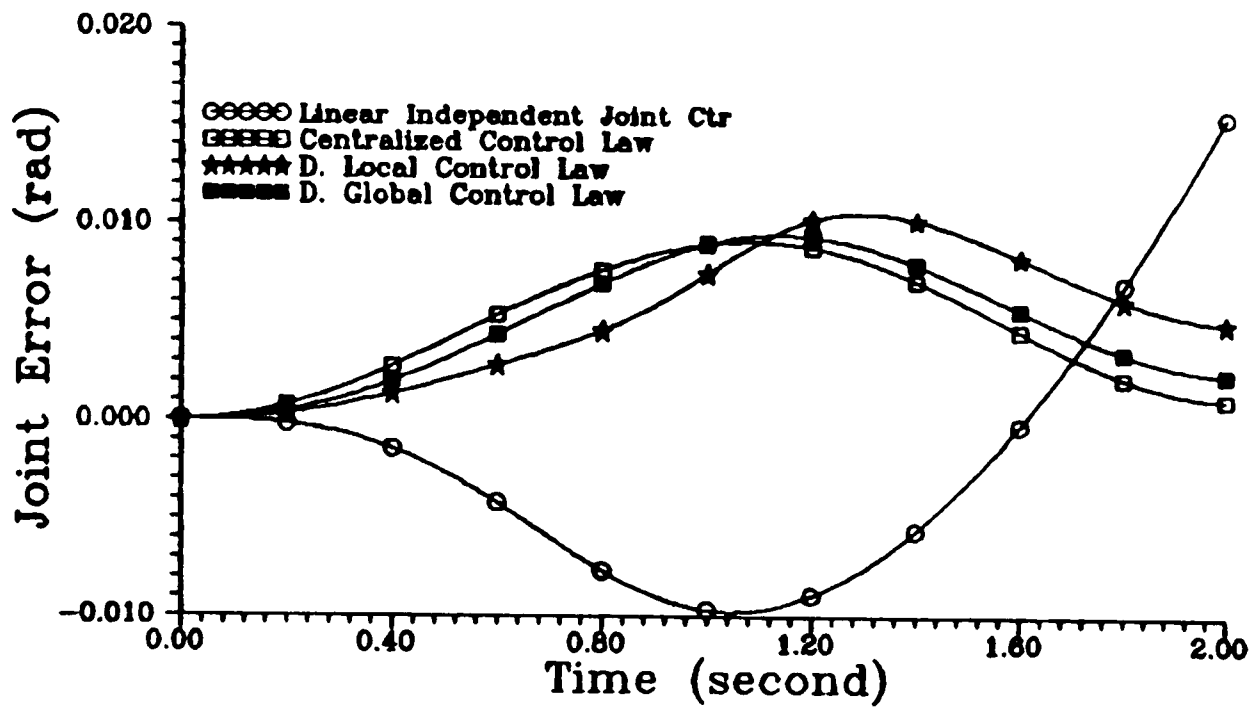


FIGURE 5.28 : Joint 3 Tracking Response Using Different Control Methods

dent joint control method. The figures also show that the decentralized control approach, especially the decentralized global control strategy, is comparable to the centralized control law.

As expected, the centralized control law gives a better tracking performance than the decentralized controllers. This is because the centralized controller was designed based on the complete robot manipulator model - without neglecting any component of the overall system model such as neglecting the off-diagonal submatrices of the overall system input matrix  $B(X, \xi, t)$ , as in the decentralized case. In the decentralized control design, one has to assume that the off-diagonal submatrices of the  $B(X, \xi, t)$  matrix are negligible compared to the diagonal submatrices in  $B(X, \xi, t)$ . This is needed so that the overall model can be decomposed into the input decentralized form, for which the decentralized control designs are based.

However, the simulation results as shown in Figures 5.29 to 5.31 do not support the above expectation for certain joints of the robot manipulator. For this simulation, the following feedback gain values were used (the values used were similar to the decentralized case for comparison purposes) :

$$K = \begin{bmatrix} -3.5781 & 111.004 & -0.065 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & -2.5606 & 127.992 & 0.247 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ & & & 0.0 & 0.0 & 0.0 \\ & & & 0.0 & 0.0 & 0.0 \\ & & & -2.5201 & 128.5255 & 0.2785 \end{bmatrix}, \quad (5.51)$$

and  $\epsilon$  has been chosen to have a value of 0.95, while  $\epsilon_1 = 0.5$  ,  $\epsilon_2 = 1.0$  ,  $\epsilon_3 = 0.5$  were used for the decentralized case. It can be seen that the magnitude of the feedback gains for this case are smaller than that of the previous case (equation 5.50).

Figures 5.29 and 5.31 show that the tracking errors using the decentralized global control law are smaller than that of the centralized approach for joints 1 and 3 of the robot manipulator; while it is the opposite for joint 2 (Figure 5.30). This is due to the fact that the  $\epsilon$  value, which determines the size of the uniform ultimate boundedness set, is the same for all the joints in the centralized controller ( $\epsilon = 0.95$  in this simulation). Whereas, in the decentralized case,  $\epsilon$  is different for each joint/subsystem ( $\epsilon_1 = 0.5$  ,  $\epsilon_2 = 1.0$  ,  $\epsilon_3 = 0.5$ ). Thus, the size of the uniform ultimate boundedness set which affect the tracking accuracy of the robot manipulator utilizing the decentralized control approach can be set differently for each joint. Comparing the value of  $\epsilon$  in the centralized case with that of the decentralized case for each joint, it can be deduced that the uniform ultimate boundedness sets for joint 1 and joint 3 are smaller for the decentralized global control than for the centralized control law. Whereas, the size of the set for joint 2 for the decentralized global control approach is bigger than that of the

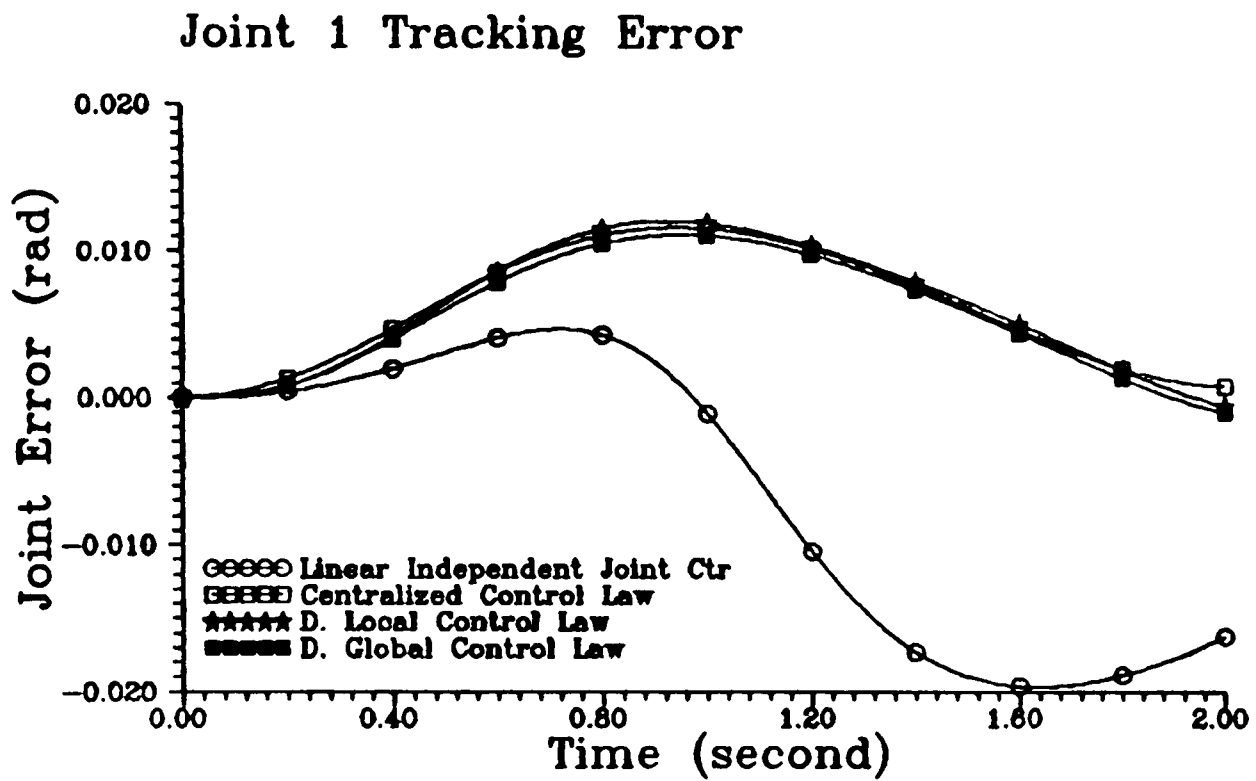


FIGURE 5.29 : Joint 1 Tracking Response Using Different Control Methods

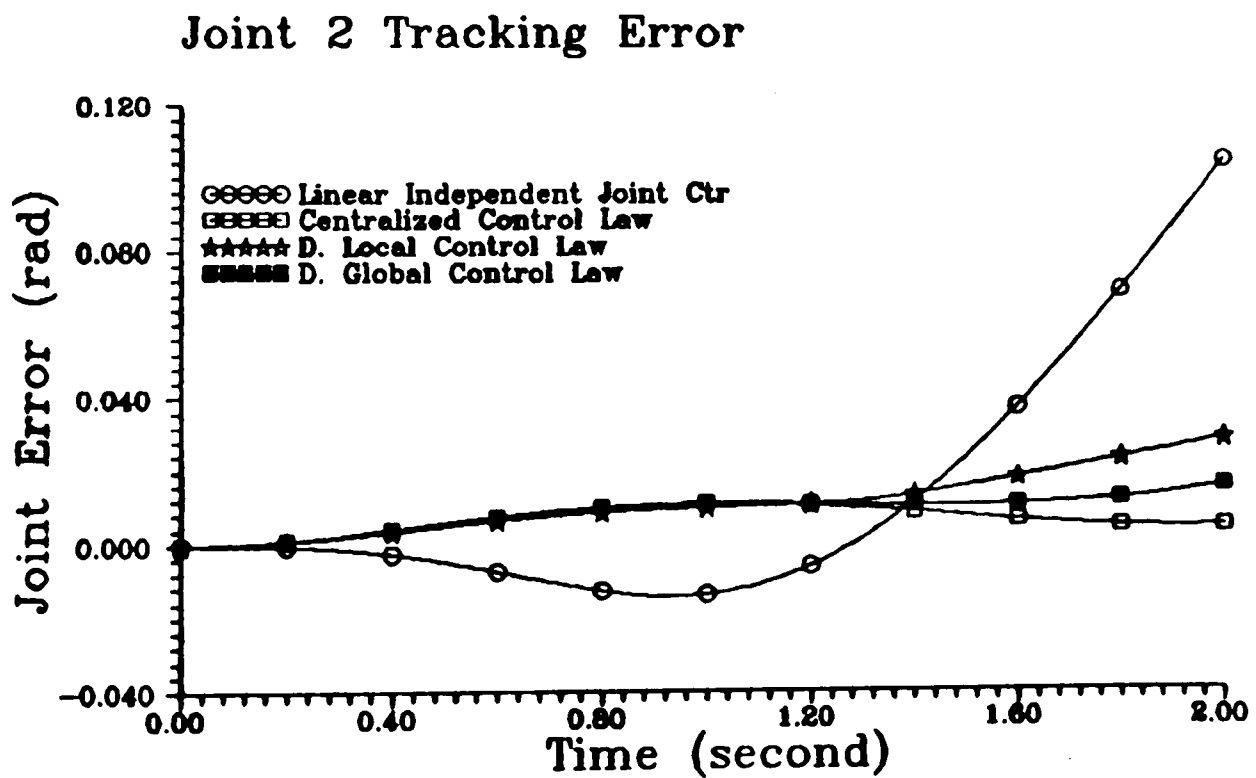


FIGURE 5.30 : Joint 2 Tracking Response Using Different Control Methods

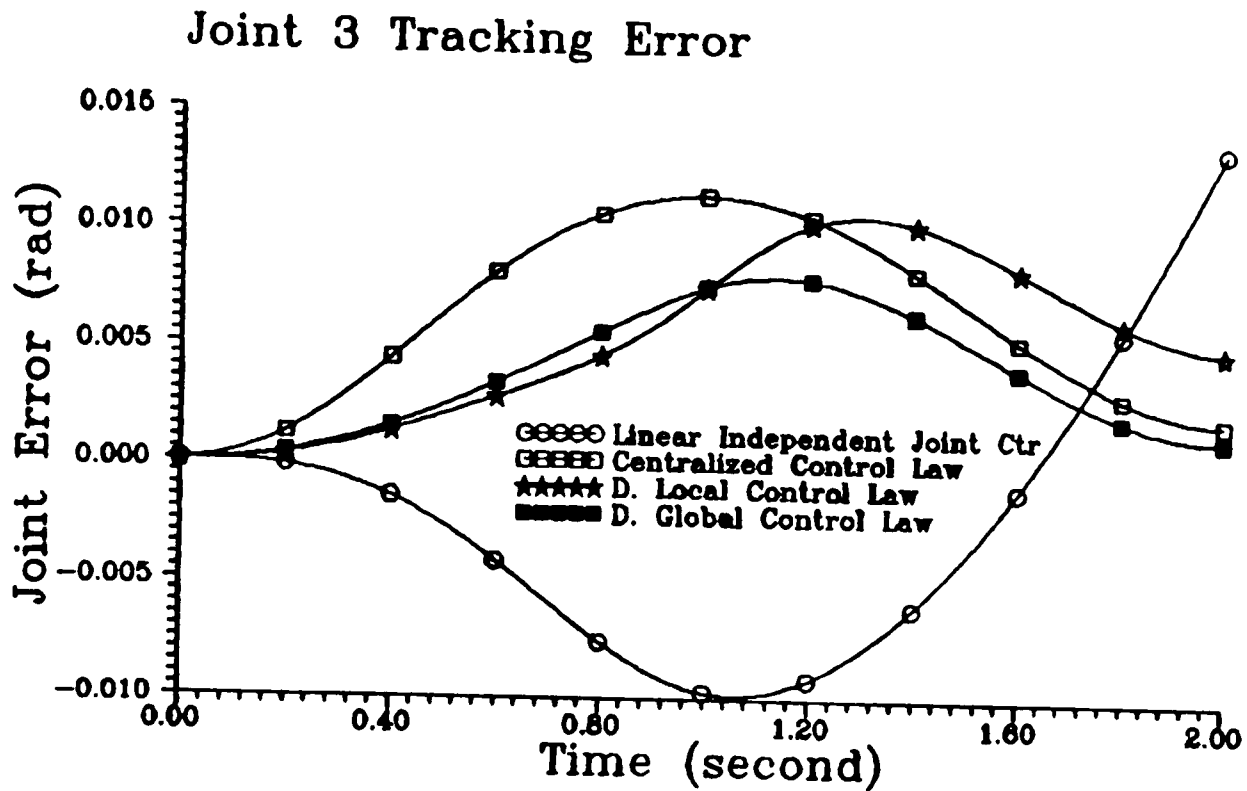


FIGURE 5.31 : Joint 3 Tracking Response Using Different Control Methods

centralized case. These findings are reflected in the simulations in the form of the tracking accuracy for the respective joints as shown in Figures 5.29 to 5.31 when the magnitude of the feedback gains used were not large enough.

The magnitude of the uncertainties, nonlinearities, and the interconnection functions may be different for each subsystem. Thus, for some subsystem, the uniform ultimate boundedness set can be made smaller than for the other subsystems. In the centralized case, the value  $\epsilon$  in the control law for each joint cannot be set independently. The same  $\epsilon$  has to be used by all the joints. This inevitably will create problems because a particular value of  $\epsilon$  might give a good tracking accuracy for a particular joint but, for another joint, chattering may result in its control signal. If the value of  $\epsilon$  is increased to overcome the chattering in the affected joint, the tracking error for the original joint will inevitably increase. Thus, it is important to have the flexibility in setting the value of  $\epsilon$  for each joint (subsystem) independently from the other subsystems.

Due to this fact, the decentralized global control law, or even the decentralized local control approach, is better than the centralized control law considered if the magnitude of the feedback gains used are not large enough.

The other advantage of the decentralized global control law, compared to the centralized control approach, is that the decentralized control law is simple and less time consuming to design because it is based on the decomposed models of the overall robot system which are normally a fraction of the overall order. The on-line computation of the decentralized control law is less time consuming than the centralized control approach since the latter involves a larger number of mathematical operations than the former method. This is another main advantage of the decentralized control method especially for a system with fast and nonlinear dynamics such as occurs in robot manipulators.

## 5.5 CONCLUSION

In this chapter, a decentralized tracking control algorithm based on a deterministic approach is proposed. The method uses the local states as well as the states of the neighbouring subsystems as feedback information. From the derived analysis and the simulation results, this approach is better in term of tracking precision than the decentralized local control approach as proposed in the previous chapter. The simulation results demonstrate that the proposed controller can accurately control the movement of the robotic arm with good robustness properties, and forces the robot manipulator to track the reference trajectory precisely inspite of the substantial uncertainties and nonlinearities that exist in the system. As expected, the proposed method is capable of reducing the effect of the nonlinear uncertain interconnection dynamics as compared to the decentralized local approach. The simulation results also show that the decentralized global control approach is better than the considered centralized control strategy in that the tracking accuracy of each joint can be improved independently from the rest of the robot manipulator's joints. Furthermore, the decentralized method is simple and requires less time in its design and the on-line computation time is reduced, compared to the centralized control method.

## CHAPTER 6

# HIERARCHICAL CONTROL CONCEPT IN ROBOTIC SYSTEM

### 6.1 INTRODUCTION

As an alternative to the decentralized global control approach, a hierarchical or multi-level control concept can be used to design a robust controller for robot manipulators such that the system is insensitive to nonlinearities, parameter variations and uncertainties and tracks a prespecified trajectory satisfactorily.

In applying a hierarchical or multi-level control strategy, the executive level of the robot control hierarchy is normally divided further into two levels. At the lowest level, a local decentralized tracking controller is designed for each decoupled subsystem. At the upper level, a global controller is generated in order to balance as exactly as possible the effects of the interconnections between the subsystems.

The idea of using a multi-level or hierarchical control concept for controlling a robot manipulator has been used in the literature. Bestaoui [1988] utilized a two-level hierarchical control scheme based on the Computed Torque technique to decompose the robot manipulator system and to compensate for the nonlinearities and the couplings that are present in the system. Similarly, Rößler [1980] used a two-level hierarchical control structure to control a robot manipulator. An on-line coordination algorithm is proposed to reduce the couplings between subsystems. The coordination and dynamic compensation of the local nonlinearities are done on the upper level of the hierarchical structure. At the lowest level, the subsystems are treated as linear and decoupled ones using established control methods such as an optimal control technique. Zaprjanov and

Boeva [1981] and Mikhailov et.al. [1985] proposed a two-level hierarchical decentralized control structure based on the interaction-prediction coordination principle [Singh, 1971] to generate an optimal stabilizing controller for robot manipulators. At the lowest level, independent local linear quadratic control problems are solved for each subsystem corresponding to each robot arm joint. The interactions between these subsystems are taken into account via the coordination procedure on the upper level. But before this two-level hierarchical controller is applied, the robot manipulator system is linearized first using an inverse problem technique based on the nonlinear robot manipulator dynamic equations. Then, the synthesis of the stabilizing control is carried out on the linearized model by means of the two-level hierarchical control method. In all these methods, even though the controllers are decentralized and simple to implement at the lowest level, the overall schemes are numerically involved due to the computational complexity at the intermediate stage.

In this study, robust two-level hierarchical controllers which are capable of withstanding the expected variations and uncertainties in the system are presented. The controllers are formulated based on the decentralized global approach as presented in Chapter 5. A complete model of the robot manipulator dynamics is used in designing the controllers. It is assumed that the upper bounds on the nonlinearities, couplings and uncertainties present in the system are available. The control laws are simple to implement and the numerical coordination strategies are also simple, which facilitate their use for a fast and strongly nonlinear system such as a robot manipulator.

The formulation of the two-level hierarchical tracking problem and some standard assumptions are presented in the following section.

## 6.2 PROBLEM FORMULATION

A decentralized control system can also be a hierarchical or multi-level system if the information of some subsystems depend directly on the action of other subsystems at higher or lower levels. If the decentralized global control law

(5.11) in Chapter 5 is divided into two terms  $U_i^{L1}(t)$  and  $U_i^U(t)$  which will be calculated on the first and second level, respectively, the decentralized global control system as presented in Chapter 5 will lead to a two-level hierarchical control structure. Thus, the two-level hierarchical control structures proposed in this chapter will be based on the mentioned decentralized global control law. For convenience, the interconnected robot manipulator dynamic equation and the some of the assumptions as presented in Chapter 5 are briefly reintroduced in the following.

The dynamics of a robot manipulator can be represented by a set of interconnected linear subsystems with bounded uncertainties as described by the following equations :

$$\dot{X}_i(t) = [A_i + \Delta A_i(X, \xi, t)] X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t), \quad (6.1)$$

$$X_i(t) = [ \theta_i(t), \dot{\theta}_i(t), \ddot{\theta}_i(t) ]^T, \quad (6.2)$$

$$X_i(t_o) = X_{i_o} ; \quad i \in \mathcal{I}, \quad (6.3)$$

where  $X_{i_o}$  is a given initial condition. The elements of the  $\Delta A_i(X, \xi, t)$ ,  $\Delta B_i(X, \xi, t)$ , and  $A_{ij}(X, \xi, t)$  matrices ,  $\Delta a_{ij}^i(X, \xi, t)$ ,  $\Delta b_i^i(X, \xi, t)$ , and  $a_{ij}^{ij}(X, \xi, t)$ , respectively, may be considered as uncertainties which belong to uncertainty bounding sets  $\mathcal{R}$ ,  $\mathcal{I}$ , and  $\mathcal{V}$  respectively. The uncertainty bounding sets may be defined as follows :

$$\begin{aligned} \mathcal{R} &\triangleq \{ \Delta a_{ij}^i(*) ; \forall i \in \mathcal{I}, \forall j \in \mathcal{I} \mid -r_{ij}^i \leq \Delta a_{ij}^i(*) \leq r_{ij}^i \} \\ \mathcal{I} &\triangleq \{ \Delta b_i^i(*) ; \forall i \in \mathcal{I}, \forall j \in \mathcal{I} \mid -s_i^i \leq \Delta b_i^i(*) \leq s_i^i \} \\ \mathcal{V} &\triangleq \{ a_{ij}^{ij}(*) ; \forall i \in \mathcal{I}, \forall j \in \mathcal{I} \mid \underline{a}_{ij}^{ij} \leq a_{ij}^{ij}(*) \leq \bar{a}_{ij}^{ij} \} \end{aligned} \quad (6.4)$$



where the values of  $r_{ij}^i$ ,  $s_i^i$ ,  $\underline{a}_{ij}^{ij}$ , and  $\bar{a}_{ij}^{ij}$  are as given in (3.13) and (3.17).

Let a continuous function  $X_d(t) \in \mathbb{R}^{3N}$  be the desired state trajectory, where  $X_d(t)$  is defined as :

$$X_d(t) = [ X_{d1}^T(t), X_{d2}^T(t), \dots, X_{dN}^T(t) ]^T$$

$$X_{di}(t) = [ \theta_{di}(t), \dot{\theta}_{di}(t), \ddot{\theta}_{di}(t) ]^T. \quad (6.5)$$

$$X_{di}(t) \in \mathbb{R}^3 ; \quad i \in \mathcal{I}.$$

It is assumed that :

- i. The pair  $(A_i, B_i)$  is stabilizable,
- ii. There exist continuous functions  $H_i(X, \xi, t) \in \mathbb{R}^{1 \times 3}$ ,  $E_i(X, \xi, t) \in \mathbb{R}$ , and  $G_{ij}(X, \xi, t) \in \mathbb{R}^{1 \times 3}$ , such that for all  $X \in \mathbb{R}^N$  and all  $t$  :

$$\Delta A_i(X, \xi, t) = B_i H_i(X, \xi, t) \quad (6.6)$$

$$\Delta B_i(X, \xi, t) = B_i E_i(X, \xi, t) \quad (6.7)$$

$$G_{ij}(X, \xi, t) = B_i G_{ij}(X, \xi, t). \quad (6.8)$$

$$\|E_i(X, \xi, t)\| < 1. \quad (6.9)$$

- iii. There exist a Lebesgue function  $\Omega_i(t) \in \mathbb{R}$ , which is integrable on bounded intervals, such that

$$\dot{X}_{di}(t) = A_i X_{di}(t) + B_i \Omega_i(t). \quad (6.10)$$

The explanations for the assumptions are omitted here since they can be found in Chapters 4 and 5.

For each subsystem, let the state tracking error between the actual and the desired states be defined as  $Z_i(t)$ , that is,

$$Z_i(t) = X_i(t) - X_{di}(t) \quad (6.11)$$

and,

$$Z_i(t_o) = X_i(t_o) - X_{di}(t_o) = Z_{io} . \quad (6.12)$$

Using (6.1) and (6.10), the error state equation for the  $i$ th subsystem can be obtained as follows :

$$\begin{aligned} \dot{Z}_i(t) = & A_i Z_i(t) + \Delta A_i(X, \xi, t) X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i(t) - \\ & B_i \Omega_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t) . \end{aligned} \quad (6.13)$$

The error state equation for the overall system may be obtained as follows :

$$\dot{Z}(t) = AZ(t) + A(X, \xi, t)X(t) - AX(t) + B(X, \xi, t)U(t) - B\Omega(t), \quad (6.14)$$

where

$$A = \text{diag} [ A_1, A_2, \dots, A_N ]$$

$$B = \text{diag} [ B_1, B_2, \dots, B_N ]$$

$$Z(t) = [ Z_1^T(t), Z_2^T(t), \dots, Z_N^T(t) ]^T$$

$$\Omega(t) = [ \Omega_1(t), \Omega_2(t), \dots, \Omega_N(t) ]^T.$$

The objective is then to design a robust two-level hierarchical control law  $U_i(t) = U_i^L(t) + U_i^U(t)$ , where  $U_i^L(t)$  and  $U_i^U(t)$  are generated on the first and second levels respectively, such that, for any uncertainties, interconnection functions, and initial conditions, the overall error system (eqn. (6.14)) with the control law

$$U(t) = [ U_1(t), U_2(t), \dots, U_N(t) ]^T, \quad (6.15)$$

is practically stable.

Associated with the problem is the 'decomposition' of the interconnected robot manipulator dynamic equation (6.1) (hence, the error equation 6.13) between the two levels in order to generate the control signals  $U_i^L(t)$  and  $U_i^U(t)$  respectively. In other words, the interconnected robot manipulator equation should be decomposed into two components : one component corresponds to the first level to generate  $U_i^L(t)$ , and the other component corresponds to the second level to produce the coordinator  $U_i^U(t)$ . There are two ways for decomposing the integrated robot manipulator equation corresponding to the levels; thus, two methods can be proposed to derive a two-level hierarchical control law for the robot manipulator system. These are as follows :

In the first method, which will be presented in the next section, the global controller  $U_i^U(t)$  at the second level is used to compensate for all the uncertainties, the nonlinearities, and the interconnections between the subsystems. It is assumed that all the upper bounds for the uncertainties and the interconnections are available. The controller is formulated based on a deterministic approach, and is nonlinear in structure. At the first level, the local decentralized controller  $U_i^L(t)$  is designed based on a decoupled linear nominal model of the subsystem. Thus, the controller  $U_i^L(t)$  in this method is linear.

In the second method, the local decentralized controller  $U_i^L(t)$  is designed using a deterministic approach based on the decoupled uncertain model of the interconnected uncertain robot manipulator equation. That is, the controller is designed ignoring the interconnection functions. The local controller  $U_i^L(t)$  is nonlinear in structure. It is assumed that the upper bounds of the uncertainties present in the decoupled subsystem model are available. At the second level, a global controller  $U_i^U(t)$  is generated taking into account the effect of the interconnection functions. Similarly, the global controller  $U_i^U(t)$  is designed based on a deterministic approach and is nonlinear in structure. This method will be outlined in Section 6.4.

In the following, superscripts 1 and 2 are introduced in  $U_i^L(t)$  and  $U_i^U(t)$  to indicate that the controllers are designed based on the first method and the second method, respectively. The vector norms are Euclidean, and the matrix norms are the corresponding induced one, that is,  $\|A\| = \sqrt{\lambda_{\max}(A^T A)}$ , where

$\lambda_{\max}(\mathbb{A}^T \mathbb{A})$  denotes the maximum eigenvalue of the matrix  $\mathbb{A}^T \mathbb{A}$ .

### 6.3 HIERARCHICAL CONTROL STRUCTURE – METHOD 1

In this approach, the coordinator  $U_i^{U1}(t)$  at the upper level is asked to compensate directly all the nonlinearities, the uncertainties and the interactions between the subsystems as exactly as possible using a deterministic approach. At the lower level, the local controllers  $U_i^{L1}(t)$  are designed ignoring the uncertainties and the interconnections. That is, the subsystems are treated as if they are decoupled and linear time invariant systems.

#### 6.3.1 Controller Design At The Lowest Level – Linear Feedback Control

At this level, it is feasible to choose arbitrary control strategies for the subsystems on the condition that the local decoupled free control subsystems are stable. Here, the local subsystem is treated as a servomechanism problem in which it is desired to find a linear control law in such a way as to cause the state of the subsystem to track or follow a desired state trajectory asymptotically.

The free decoupled subsystem model is given by the state equation :

$$\dot{X}_i(t) = A_i X_i(t) + B_i U_i^{L1}(t). \quad (6.16)$$

Using (6.10) and (6.11), the error dynamics for each decoupled free subsystem is given as follows :

$$\dot{Z}_i(t) = A_i Z_i(t) + B_i [ U_i^{L1}(t) - \Omega_i(t) ]. \quad (6.17)$$

For each decoupled free subsystem, the following linear decentralized local controller is used :

$$U_i^{L1}(t) = \tilde{U}_i(Z_i, t) = K_i Z_i(t) + \Omega_i(t) . \quad (6.18)$$

It can be shown easily that the decoupled free subsystem is asymptotically stable and tracks the desired trajectory if the closed loop subsystem matrix

$$\tilde{A}_i \triangleq A_i + B_i K_i \quad (6.19)$$

is a stable matrix. This can be achieved by selecting the feedback matrix  $K_i$  such that all the eigenvalues of the matrix  $\tilde{A}_i$  are in the left half of the complex plain.

Figure 6.1 shows the block diagram of the  $i$ th subsystem at the first level for this method.

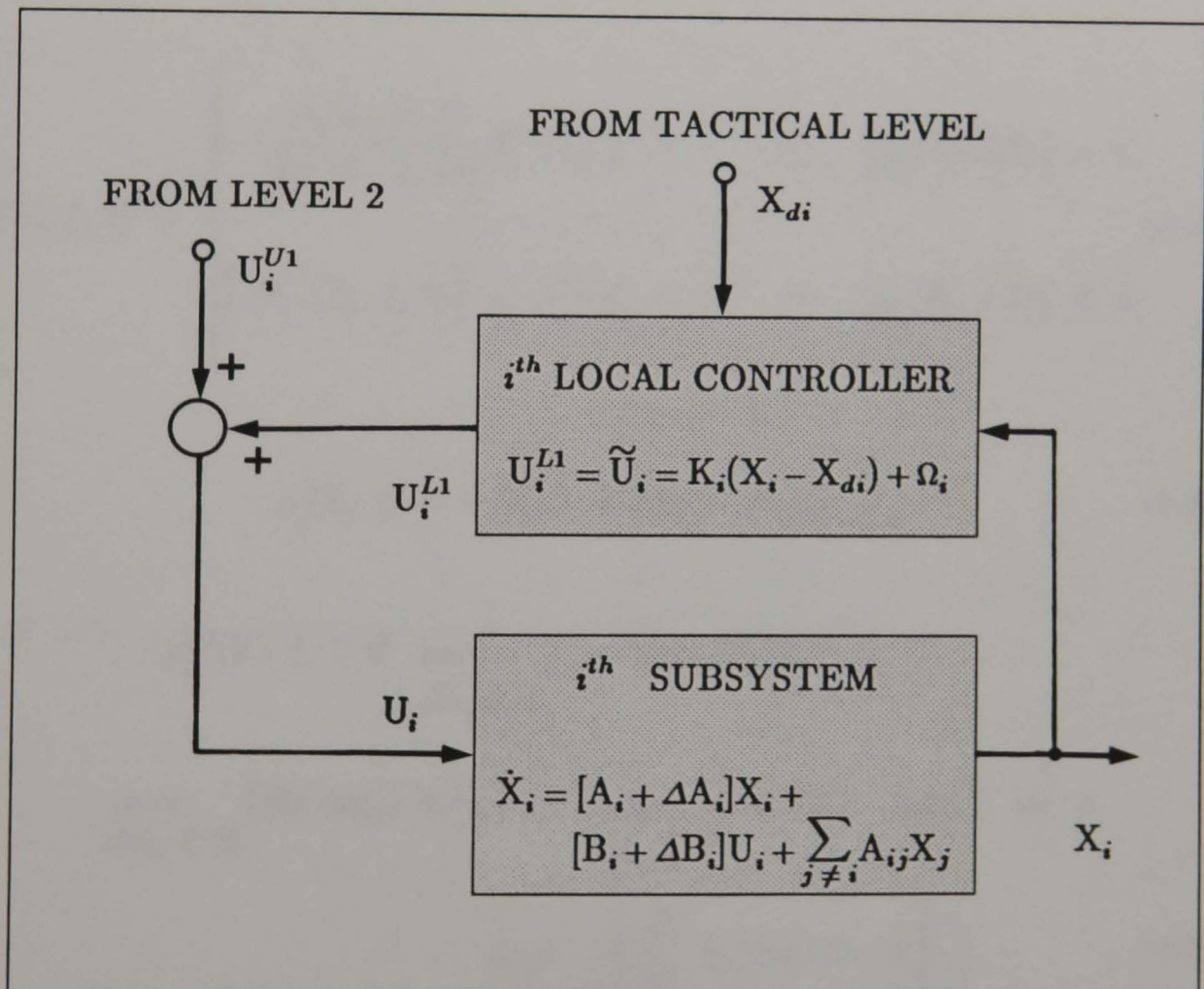


FIGURE 6.1 : Decentralized Linear Controller For  $i$ th Decoupled Subsystem At The Lowest Level Using Method 1

### 6.3.2 Controller Design At The Upper Level – Deterministic Approach

In this level, the coordinator takes into account all the system nonlinearities, uncertainties and interconnections between the subsystems based on their maximum bounds. Unlike in the first method where the coordinator is only used to overcome the influence of the interconnections between the subsystems, the aim of the coordinator in this method is to compensate the influence of all the nonlinearities, the uncertainties and the interconnections present in the system.

In this method, the following coordinator is proposed :

$$U_i^{U1}(t) = \Phi_i^{U1}(Z_i, \xi, t), \quad (6.20)$$

where

$$\Phi_i^{U1}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_i(Z_i, \xi, t)}{\|\mu_i(Z_i, \xi, t)\|} \rho_i^{U1}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| > \epsilon_i \\ \|\Phi_i^{U1}(Z_i, \xi, t)\| \leq \rho_i^{U1}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (6.21)$$

$$\mu_i(Z_i, \xi, t) = B_i^T P_i Z_i(t) \rho_i^{U1}(Z_i, \xi, t) \quad (6.22)$$

$$\rho_i^{U1}(Z_i, \xi, t) \triangleq \max_{\Delta b_i^i \in \mathcal{Y}} [1 - \|E_i(\Delta b_i^i)\|]^{-1} \left( \max_{\Delta a_{ij}^i \in \mathcal{R}} \|H_i(\Delta a_{ij}^i) X_i(t)\| + \max_{\Delta b_i^i \in \mathcal{Y}} \|E_i(\Delta b_i^i) \tilde{U}_i(Z_i, \xi, t)\| + \max_{a_{ij}^{ij} \in \mathcal{V}} \left\| \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij}(a_{ij}^{ij}) X_j(t) \right\| \right), \quad (6.23)$$

and  $\epsilon_i$  in equation (6.21) is a prescribed positive constant.  $P_i$  is the solution of the

matrix Lyapunov equation

$$P_i \tilde{A}_i + \tilde{A}_i^T P_i = -Q_i, \quad (6.24)$$

for a given positive definite symmetric matrix  $Q_i$ .

It can be seen that equation (6.23) is the same as equation (5.14). Thus, all the uncertainties, the nonlinearities and the interconnections between the subsystem are being compensated by the nonlinear controller (6.20) from this level. Hence, the introduction of the second level is useful in providing a robust controller for the robot manipulator system. Figure 6.2 illustrates the proposed two-level hierarchical control concept presented in this section.

### Theorem 6.1

The set of interconnected system (6.1) satisfying all the assumptions (i) - (iii), is practically stable via the two-level hierarchical control law

$$U(t) = [U_1(t), U_2(t), \dots, U_N(t)] \quad (6.25)$$

$$U_i(t) = U_i^{L1}(t) + U_i^{U1}(t), \quad (6.26)$$

where  $U_i^{L1}(t)$  and  $U_i^{U1}(t)$  are as given in equations (6.18) and (6.20), respectively, and tracks the reference trajectories (6.5) to within any neighbourhood of  $\mathfrak{B}(\eta_{H1})$ , where

$$\eta_{H1} = \sqrt{\frac{\sum_i^N 4\epsilon_i}{\lambda_{\min}(\mathbb{T}_{H1})}}. \quad (6.27)$$

### Proof

The proof is similar to the proof of theorem 5.1 in Section 5.3. Thus, it is omitted here.



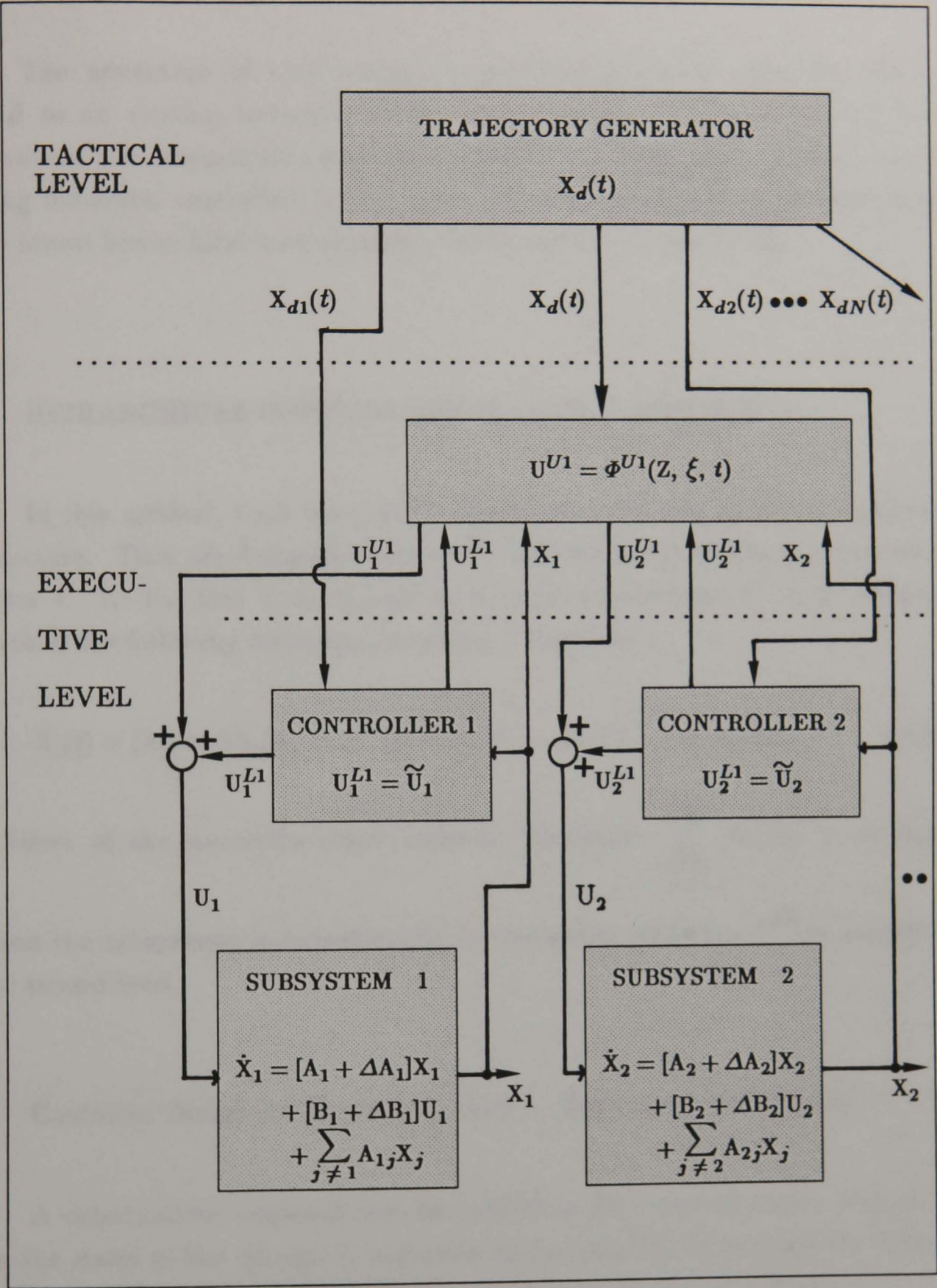


FIGURE 6.2 : Two-Level Hierarchical Control Of Robot Manipulator – Method 1



The advantage of this method is that the proposed controller can be applied to an existing industrial robot using a proportional plus derivative or proportional plus integral plus derivative controller for each joint. In this case the existing industrial controllers for the industrial robot manipulator are considered as the lowest hierarchical level replacing the linear control law (6.18).

## 6.4 HIERARCHICAL CONTROL STRUCTURE – METHOD 2

In this method, both the control signals from the two levels are nonlinear in structure. They are designed based on the deterministic approach presented in Chapter 4. At the first level, a local decentralized controller  $U_i^{L2}(t)$  is designed for each of the following decoupled uncertain subsystems :

$$\dot{X}_i(t) = [A_i + \Delta A_i(X, \xi, t)] X_i(t) + [B_i + \Delta B_i(X, \xi, t)] U_i^{L2}(t). \quad (6.28)$$

The effect of the uncertain interconnection functions  $\sum_{\substack{j=1 \\ j \neq i}}^N A_{ij}(X, \xi, t) X_j(t)$

between the subsystems is accounted for by the global controller  $U_i^{U2}(t)$  generated at the second level.

### 6.4.1 Controller Design At The Lowest Level – Deterministic Approach

A deterministic approach can be applied at this level in such a way as to cause the states of the decoupled uncertain subsystems (6.28) to track the desired states irrespective of the uncertainties present in each decoupled subsystem.

Using assumptions (ii) and (iii), the error dynamics for each decoupled uncertain subsystem can be derived as follows :

$$\dot{Z}_i(t) = A_i Z_i(t) + B_i \left\{ U_i^{L2}(t) + H_i(X, \xi, t) X_i(t) + E_i(X, \xi, t) U_i^{L2}(t) - \Omega_i(t) \right\}. \quad (6.29)$$

The objective is then to design the local decentralized control law  $U_i^{L2}(t)$  such that the error system (6.29) is practically stable and tracks the desired trajectory  $X_{di}(t)$ .

In this method, the following decentralized control law is proposed for each decoupled subsystem :

$$U_i^{L2}(t) = \tilde{U}_i(Z_i, t) + \Phi_i^{L2}(Z_i, \xi, t), \quad (6.30)$$

where  $\tilde{U}_i(Z_i, t)$  is as given in equation (6.18), and

$$\Phi_i^{L2}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_i(Z_i, \xi, t)}{\|\mu_i(Z_i, \xi, t)\|} \rho_i^{L2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| > \epsilon_i \\ \|\Phi_i^{L2}(Z_i, \xi, t)\| \leq \rho_i^{L2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (6.31)$$

$$\mu_i(Z_i, \xi, t) = B_i^T P_i Z_i(t) \quad (6.32)$$

$$\rho_i^{L2}(Z_i, \xi, t) \triangleq \max_{\Delta b_i^i \in \mathcal{Y}} [1 - \|E_i(\Delta b_i^i)\|]^{-1} \left( \max_{\Delta a_{ij}^i \in \mathcal{R}} \|H_i(\Delta a_{ij}^i) X_i(t)\| + \max_{\Delta b_i^i \in \mathcal{Y}} \|E_i(\Delta b_i^i) \tilde{U}_i(Z_i, \xi, t)\| \right), \quad (6.33)$$

and  $\epsilon_i$  and  $P_i$  are as defined earlier. The constant gain matrix  $K_i$  is chosen, such that, the closed loop system matrix  $\tilde{A}_i \triangleq A_i + B_i K_i$  is asymptotically stable. The structure of the decentralized local controller can be illustrated as in Figure 6.3.

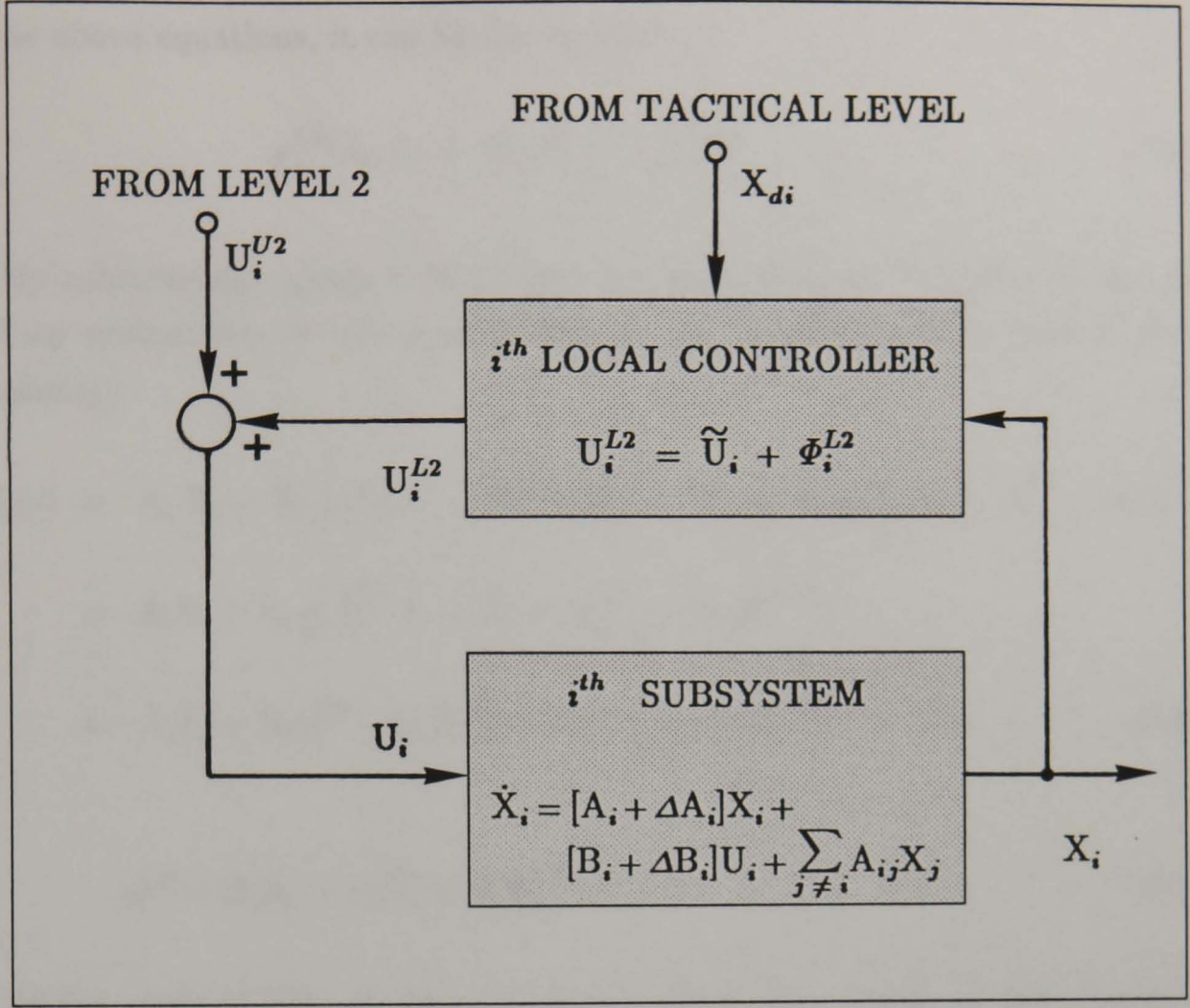


FIGURE 6.3 : Decentralized Nonlinear Controller For The  $i$ th Decoupled Subsystem At The Lowest Level Using Method 2

Define :

$$\gamma_{1i}^L \triangleq \max_{\Delta b_i^i \in \mathcal{Y}} \left( 1 - \|E_i(\Delta b_i^i)\| \right)^{-1} \left( \max_{\Delta a_{ij}^i \in \mathcal{R}} \|H_i(\Delta a_{ij}^i) X_i(t)\| + \max_{\Delta b_i^i \in \mathcal{Y}} \|E_i(\Delta b_i^i) \Omega_i\| \right), \quad (6.34)$$

$$\gamma_{2i}^L \triangleq \max_{\Delta b_i^i \in \mathcal{Y}} \left( 1 - \|E_i(\Delta b_i^i)\| \right)^{-1} \left( \max_{\Delta b_i^i \in \mathcal{Y}} \|E_i(\Delta b_i^i) K_i\| \right). \quad (6.35)$$

From the above equations, it can be shown that

$$\rho_i^{L2}(Z_i, \xi, t) \leq \gamma_{1i}^L + \gamma_{2i}^L \| Z_i \| \quad . \quad (6.36)$$

By substituting equation (6.30) into the error dynamic equation (6.29), the closed loop system can be obtained as follows (the arguments have been omitted for simplicity) :

$$\begin{aligned} \dot{Z}_i(t) &= A_i Z_i + B_i \left\{ K_i Z_i + \Omega_i + \Phi_i^{L2} + H_i X_i + E_i \tilde{U}_i + E_i \Phi_i^{L2} - \Omega_i \right\} \\ &= \tilde{A}_i Z_i + B_i \left\{ \Phi_i^{L2} + H_i X_i + E_i \tilde{U}_i + E_i \Phi_i^{L2} \right\} \\ &= \tilde{A}_i Z_i + B_i e_i^{L2} + B_i \Phi_i^{L2}, \end{aligned} \quad (6.37)$$

where

$$e_i^{L2} = H_i X_i + E_i \tilde{U}_i + E_i \Phi_i^{L2} \quad . \quad (6.38)$$

Using (6.31) and (6.33), it can be shown that the norm of the 'lumped' uncertainties  $e_i^{L2}$  for the decoupled subsystem is bounded as follows :

$$\begin{aligned} \| e_i^{L2} \| &\leq \| H_i X_i \| + \| E_i \tilde{U}_i \| + \| E_i \| \rho_i^{L2} \\ &\leq \rho_i^{L2} \quad . \end{aligned} \quad (6.39)$$

Using the following Lyapunov function

$$\mathcal{L}_i(Z_i, t) = Z_i^T P_i Z_i, \quad (6.40)$$

and equations (6.36) and (6.39), by following the same procedure as in Chapters 4 and 5, it can be shown that the derivative of the Lyapunov functions along the error equation (6.37) for each of the decoupled system can be obtained as follows :

$$\dot{\mathbf{z}}_i(\mathbf{Z}_i, t) \leq -\lambda_{\min}(\mathbf{Q}_i) \|\mathbf{Z}_i\|^2 + 4\epsilon_i \gamma_{2i}^L \|\mathbf{Z}_i\| + 4\epsilon_i \gamma_{1i}^L . \quad (6.41)$$

It can be seen that (6.41) is of the form of equation (4.6) in section 4.2.1, thus, if  $\lambda_{\min}(\mathbf{Q}_i) > 0$ ,  $\dot{\mathbf{z}}_i(\mathbf{Z}_i, t) < 0$  for all  $t$  and  $\mathbf{Z}_i(t) \in \mathfrak{B}^c(\eta_{Li})$ , where  $\mathfrak{B}^c(\eta_{Li})$  is the complement of the closed ball  $\mathfrak{B}(\eta_{Li})$ , centred at  $\mathbf{Z}_i(t)=0$  with radius

$$\eta_{Li} = \frac{2 \epsilon_i \gamma_{2i}^L}{\lambda_{\min}(\mathbf{Q}_i)} + \left[ \left( \frac{\epsilon_i \gamma_{2i}^L}{\lambda_{\min}(\mathbf{Q}_i)} \right)^2 + \frac{4 \epsilon_i \gamma_{1i}^L}{\lambda_{\min}(\mathbf{Q}_i)} \right]^{\frac{1}{2}}, \quad (6.42)$$

and the decoupled error subsystem (6.29) with the nonlinear decentralized controller (6.30), is uniformly ultimately bounded with respect to the set  $\mathfrak{X}(\underline{\kappa}_i)$ , where

$$\mathfrak{X}(\underline{\kappa}_i) \triangleq \left\{ \mathbf{Z}_i \in \mathbb{R}^N \mid \mathbf{Z}_i^T \mathbf{P}_i \mathbf{Z}_i < \underline{\kappa}_i \right\}, \quad (6.43)$$

$$\underline{\kappa}_i = \lambda_{\max}(\mathbf{P}_i) \eta_{Li}^2, \quad (6.44)$$

Thus, at the lowest level, each decoupled uncertain subsystem is practically stable under the decentralized local control law (6.30).

#### 6.4.2 Controller Design At The Upper Level – Deterministic Approach

The objective of this level is to compensate for the effect of the interconnections between the subsystems. In order to do that, the following nonlinear control law is proposed :

$$\mathbf{U}_i^{U2}(t) = \boldsymbol{\Phi}_i^{U2}(\mathbf{Z}_i, \boldsymbol{\xi}, t), \quad (6.45)$$

where

$$\Phi_i^{U2}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_i(Z_i, \xi, t)}{\|\mu_i(Z_i, \xi, t)\|} \rho_i^{U2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| > \epsilon_i \\ \|\Phi_i^{U2}(Z_i, \xi, t)\| \leq \rho_i^{U2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (6.46)$$

$$\rho_i^{U2}(Z_i, \xi, t) \triangleq \max_{\Delta b_i \in \mathcal{Y}} \left( 1 - \|\mathbb{E}_i(\Delta b_i)\| \right)^{-1} \left( \sum_{\substack{j=1 \\ j \neq i}}^N \max_{a_{ij}^{ij} \in \mathcal{V}} \|\mathbf{G}_{ij}(a_{ij}^{ij}) X_j(t)\| \right), \quad (6.47)$$

and  $\mu_i(Z_i, \xi, t)$  and  $\epsilon_i$  are as defined earlier.

It can be seen from equation (6.47) that the interconnections between the subsystem is compensated by the nonlinear controller (6.45) from this level. Next, the practical stability of the overall system with the proposed two-level hierarchical control concept will be analyzed.

### 6.4.3 Stability Of The Overall System

#### Theorem 6.2

The overall error system (6.14) is practically stable via the two-level hierarchical control law

$$\mathbf{U}(t) = [U_1(t), U_2(t), \dots, U_N(t)] \quad (6.48)$$

$$U_i(t) = U_i^{L2}(t) + U_i^{U2}(t), \quad (6.49)$$

where  $U_i^{L2}(t)$  and  $U_i^{U2}(t)$  are as given in equations (6.30) and (6.45), respectively, and tracks the reference trajectories (6.5) to within any neighbourhood of  $\mathfrak{B}(\eta_{H2})$ , where

$$\eta_{H2} = \frac{\|\mathbf{V}_{H2}\|}{2\lambda_{\min}(\mathbb{T}_{H2})} + \left[ \left( \frac{\|\mathbf{V}_{H2}\|}{4\lambda_{\min}(\mathbb{T}_{H2})} \right)^2 + \frac{\sum_i^N 4\gamma_{1i}\epsilon_i}{\lambda_{\min}(\mathbb{T}_{H2})} \right]^{\frac{1}{2}}, \quad (6.50)$$

where  $\mathbb{T}_{H2}$  is an  $N \times N$  the positive definite matrix, and  $\mathbb{V}_{H2}$  is an  $N \times N$  diagonal matrix to be defined later.

### **Proof**

Let,

$$\Phi_i(Z_i, \xi, t) = \Phi_i^{L2}(Z_i, \xi, t) + \Phi_i^{U2}(Z_i, \xi, t) \quad (6.51)$$

$$\rho_i(Z_i, \xi, t) = \rho_i^{L2}(Z_i, \xi, t) + \rho_i^{U2}(Z_i, \xi, t). \quad (6.52)$$

Furthermore, from (6.31) and (6.46),

$$\| \Phi_i^{L2}(Z_i, \xi, t) \| \leq \rho_i^{L2}(Z_i, \xi, t) \quad (6.53)$$

$$\| \Phi_i^{U2}(Z_i, \xi, t) \| \leq \rho_i^{U2}(Z_i, \xi, t). \quad (6.54)$$

Then,

$$\begin{aligned} \| \Phi_i(Z_i, \xi, t) \| &= \| \Phi_i^{L2}(Z_i, \xi, t) + \Phi_i^{U2}(Z_i, \xi, t) \| \\ &\leq \| \Phi_i^{L2}(Z_i, \xi, t) \| + \| \Phi_i^{U2}(Z_i, \xi, t) \| \\ &\leq \rho_i^{L2}(Z_i, \xi, t) + \rho_i^{U2}(Z_i, \xi, t) \\ &= \rho_i(Z_i, \xi, t) \end{aligned} \quad (6.55)$$

Substituting the control law  $U_i(t) = U_i^{L2}(t) + U_i^{U2}(t)$  into the error equation (6.14), and using assumptions (ii) gives :

$$\begin{aligned} \dot{Z}_i(t) &= A_i Z_i - B_i \Omega_i + B_i [H_i X_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j] + B_i [I + E_i] [U_i^{L2} + U_i^{U2}] \\ &= \tilde{A}_i Z_i - B_i [K_i Z_i + \Omega_i] + B_i [H_i X_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j] + \end{aligned}$$

$$\begin{aligned}
& B_i[I+E_i][\tilde{U}_i + \Phi_i^{L2} + \Phi_i^{U2}] \\
&= \tilde{A}_i Z_i + B_i[H_i X_i + E_i \tilde{U}_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j] + B_i[I+E_i] [\Phi_i^{L2} + \Phi_i^{U2}] \\
&= \tilde{A}_i Z_i + B_i e_i + B_i \Phi_i,
\end{aligned} \tag{6.56}$$

where

$$e_i = H_i X_i + E_i \tilde{U}_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j + E_i \Phi_i. \tag{6.57}$$

Using (6.51), it can be shown that the 'lumped' uncertainties represented by equation (6.57) is composed of two components :

$$\begin{aligned}
e_i &= H_i X_i + E_i \tilde{U}_i + \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j + E_i (\Phi_i^{L2} + \Phi_i^{U2}) \\
&= (H_i X_i + E_i \tilde{U}_i + E_i \Phi_i^{L2}) + \left( \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j + E_i \Phi_i^{U2} \right) \\
&= e_i^{L2} + e_i^{U2},
\end{aligned} \tag{6.58}$$

where  $e_i^{L2}$  is as defined in equation (6.38), and

$$e_i^{U2} = \sum_{\substack{j=1 \\ j \neq i}}^N G_{ij} X_j + E_i \Phi_i^{U2}. \tag{6.59}$$

From (6.47),

$$\| e_i^{U2} \| \leq \sum_{\substack{j=1 \\ j \neq i}}^N \| G_{ij} X_j \| + \| E_i \Phi_i^{U2} \| \leq \rho_i^{U2}. \tag{6.60}$$

Then, from (6.39), (6.60), and (6.52) :

$$\| e_i \| = \| e_i^{L2} + e_i^{U2} \|^2$$



$$\begin{aligned}
&\leq \| e_i^{L2} \| + \| e_i^{U2} \| \\
&\leq \rho_i^{L2} + \rho_i^{U2} = \rho_i .
\end{aligned} \tag{6.61}$$

Let the composite Lyapunov function be :

$$\mathcal{L}(Z, t) = \sum_i^N L_i(Z_i, t) = \sum_i^N Z_i^T P_i Z_i . \tag{6.62}$$

The derivative of the Lyapunov function with respect to time  $t$  can be obtained as follows :

$$\dot{\mathcal{L}}(Z, t) = \sum_i^N 2 Z_i^T P_i \dot{Z}_i . \tag{6.63}$$

Substituting equation (6.56) into (6.63), gives :

$$\begin{aligned}
\dot{\mathcal{L}}(Z, t) &= \sum_i^N 2 Z_i^T P_i \tilde{A}_i Z_i + \sum_i^N 2 Z_i^T P_i B_i \{ \Phi_i + e_i \} \\
&= \sum_i^N 2 Z_i^T P_i \tilde{A}_i Z_i + \sum_i^N 2 (B_i^T P_i Z_i)^T \{ \Phi_i + e_i \} .
\end{aligned} \tag{6.64}$$

Following the similar procedure as in Section 4.4, the derivative of the composite Lyapunov function along the error system (6.56) can be obtained as the following:

$$\begin{aligned}
\dot{\mathcal{L}}(Z, t) &\leq - \sum_i^N \lambda_{\min}(Q_i) \| Z_i \|^2 + \sum_i^N 4 \gamma_{2i} \epsilon_i \| Z_i \| + \sum_i^N 4 \gamma_{1i} \epsilon_i \\
&\leq - \tilde{Z}^T \mathbb{T}_{H2} \tilde{Z} + \mathbb{V}_{H2} \tilde{Z} + \sum_i^N 4 \gamma_{1i} \epsilon_i \\
&\leq - \lambda_{\min}(\mathbb{T}_{H2}) \| \tilde{Z} \|^2 + \| \mathbb{V}_{H2} \| \| \tilde{Z} \| + \sum_i^N 4 \gamma_{1i} \epsilon_i ,
\end{aligned} \tag{6.65}$$

where  $\tilde{Z}$  is as defined in equation (4.57),

$$\mathbb{T}_{H2} = \text{diag} [ \lambda_{\min}(Q_1) , \lambda_{\min}(Q_2) , \dots , \lambda_{\min}(Q_N) ] , \quad (6.66)$$

$$\mathbb{V}_{H2} = \text{diag} [ 4\gamma_{21}\epsilon_1, 4\gamma_{22}\epsilon_2, \dots, 4\gamma_{2N}\epsilon_N ] , \quad (6.67)$$

$$\gamma_{1i} \triangleq \max_{\Delta b_i^i \in \mathcal{Y}} \left( 1 - \left\| E_i(\Delta b_i^i) \right\| \right)^{-1} \left( \max_{\Delta a_{ij}^i \in \mathcal{R}} \left\| H_i(\Delta a_{ij}^i) X_i(t) \right\| + \right. \\ \left. \max_{\Delta a_{ij}^{ij} \in \mathcal{V}} \left\| G_{ij}(a_{ij}^{ij}) X_j(t) \right\| + \max_{\Delta b_i^i \in \mathcal{Y}} \left\| E_i(\Delta b_i^i) \Omega_i \right\| \right) , \quad (6.68)$$

$$\gamma_{2i} \triangleq \max_{\Delta b_i^i \in \mathcal{Y}} \left( 1 - \left\| E_i(\Delta b_i^i) \right\| \right)^{-1} \left( \max_{\Delta b_i^i \in \mathcal{Y}} \left\| E_i(\Delta b_i^i) K_i \right\| \right) . \quad (6.69)$$

Since  $\lambda_{\min}(\mathbb{T}_{H2}) > 0$  by design, this implies that  $\dot{\mathcal{L}}(Z, t) < 0$  for all  $t$  and  $\tilde{Z} \in \mathcal{B}^c(\eta_{H2})$ , where  $\mathcal{B}^c(\eta_{H2})$  is the complement of the closed ball  $\mathcal{B}(\eta_{H2})$ , centred at  $\tilde{Z}=0$  with radius  $\eta_{H2}$  as defined by equation (6.50). Hence, the error system (6.56) with the proposed two-level hierarchical control law controller  $U_i(t) = U_i^{L2}(t) + U_i^{U2}(t)$  is uniformly ultimately bounded with respect to the set  $\mathfrak{X}(\kappa_{H2})$ , where

$$\mathfrak{X}(\kappa_{H2}) \triangleq \left\{ \tilde{Z} \in \mathbb{R}^N \mid \tilde{Z}^T P \tilde{Z} < \kappa_{H2} \right\} , \quad (6.70)$$

$$\kappa_{H2} = \lambda_{\max}(P) \eta_{H2}^2 , \quad (6.71)$$

$$P = \text{diag} [ P_1 , P_2 , \dots , P_N ] . \quad (6.72)$$

This concludes the proof.  $\square$

The proposed nonlinear control law given above is slightly different from the decentralized global control (5.11) in Section 5.3 of Chapter 5 in that, the function  $\mu_i(Z_i, \xi, t)$  in equation (6.32) does not contain the norm bound of the error  $\rho_i(Z_i, \xi, t)$ . The proposed two-level hierarchical control concept is illustrated in Figure 6.4 .

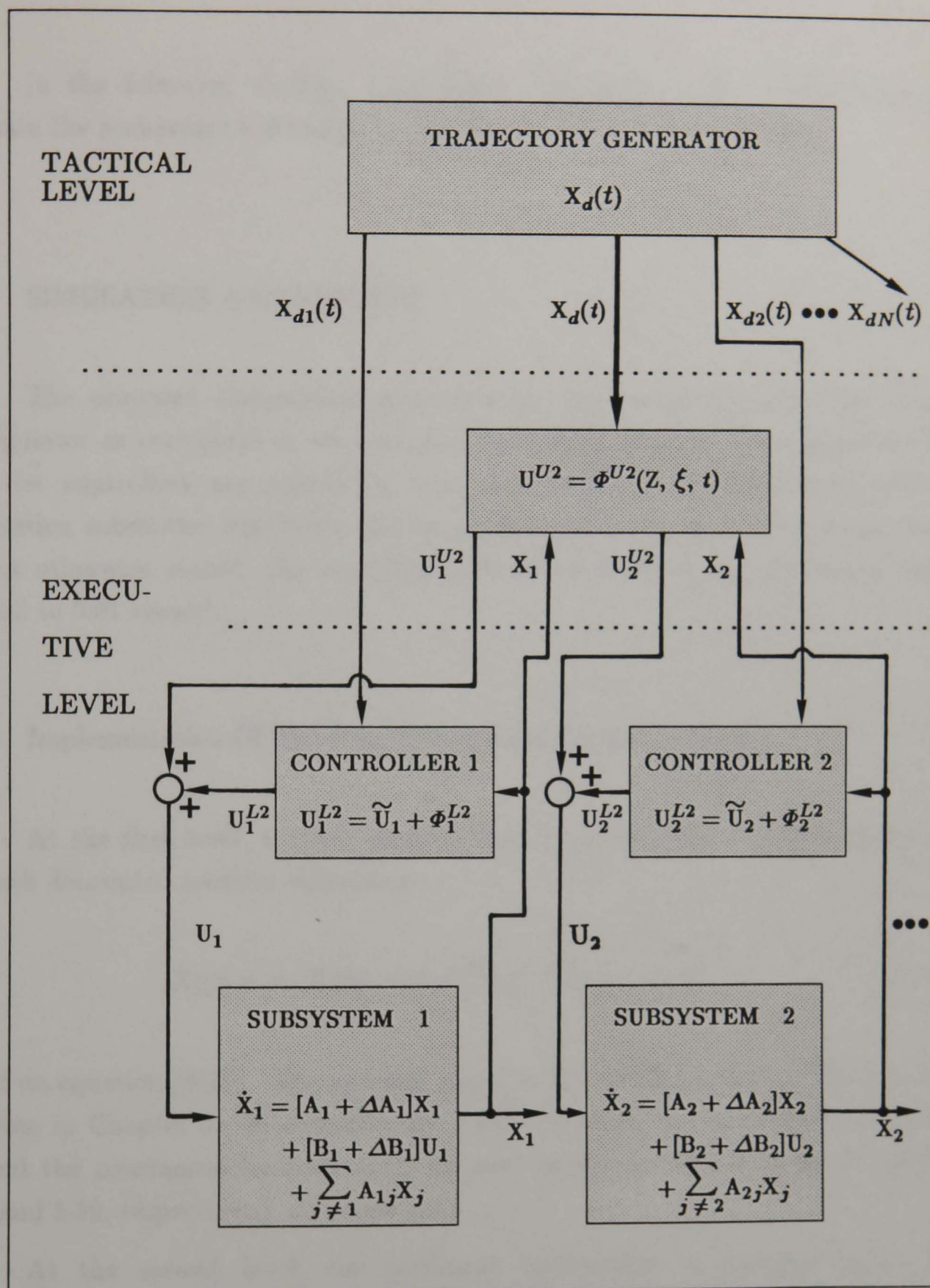


FIGURE 6.4 : Two-Level Hierarchical Control Of Robot Manipulator – Method 2 .

In the following section, a computer simulation study is conducted to evaluate the performance of the proposed hierarchical control strategies.

## 6.5 SIMULATION AND RESULTS

The proposed hierarchical controllers are applied to the three dof robot manipulator as considered in the previous simulation studies. The overall model and the controllers are solved by using the Runge-Kutta-Butcher numerical integration subroutine which has also been employed in the previous two chapters. Unless otherwise stated, the sampling interval on both of the levels has been defined to 0.01 second.

### 6.5.1 Implementation Of The First Hierarchical Control Method

At the first level, a linear decentralized local controller  $U_i^{L1}(t)$  is designed for each decoupled nominal subsystem :

$$\dot{X}_i(t) = A_i X_i(t) + B_i U_i^{L1}(t) \quad , i = 1, 2, 3, \quad (6.73)$$

based on equation (6.16). The nominal matrices  $A_i$  and  $B_i$  for each subsystem are as given in Chapter 3. In designing the local controller, the same feedback gains  $K_i$  and the continuous function  $\Omega_i(t)$  as used in the previous chapter (equation 5.40 and 5.39, respectively) are employed.

At the second level, the nonlinear coordinator is designed based on equations (6.21) to (6.23). These equations are in fact similar to equations (5.12) to (5.14), respectively. Thus, for comparison purposes, the same nonlinear controller given by equation (5.37) is used here :

$$\Phi_i^{U1}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_i(Z_i, \xi, t)}{\|\mu_i(Z_i, \xi, t)\|} \rho_i^{U1}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| > \epsilon_i \\ -\frac{\mu_i(Z_i, \xi, t)}{\epsilon_i} \rho_i^{U1}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (6.74)$$

$$\mu_i(Z_i, \xi, t) = B_i^T P_i Z_i(t) \rho_i^{L1}(Z_i, \xi, t) \quad (6.75)$$

$$\begin{aligned} \rho_1^{U1}(Z_1, \xi, t) = & 1.0339 \left\{ \|4.1174 X_2^1 + 0.5271 X_3^1\| + \|0.0328 \tilde{U}_1\| + \right. \\ & \|2.2809 X_2^2 + 0.1964 X_3^2\| + \\ & \left. \|1.3024 X_2^3 + 0.1153 X_3^3\| \right\} \end{aligned}$$

$$\begin{aligned} \rho_2^{U1}(Z_2, \xi, t) = & 1.0256 \left\{ \|28.1056 X_1^2 + 3.9371 X_2^2 + 0.1489 X_3^2\| + \right. \\ & \|0.0249 \tilde{U}_2\| + \|0.9637 X_2^1 + 0.1687 X_3^1\| + \\ & \left. \|0.2816 X_1^3 + 3.9323 X_2^3 + 0.2094 X_3^3\| \right\} \end{aligned} \quad (6.76)$$

$$\begin{aligned} \rho_3^{U1}(Z_3, \xi, t) = & 1.0003 \left\{ \|12.5947 X_1^3 + 0.3616 X_2^3 + 2.6029 \times 10^{-3} X_3^3\| + \right. \\ & \|2.6154 \times 10^{-4} \tilde{U}_3\| + \|0.579 X_2^1 + 0.1016 X_3^1\| + \\ & \left. \|0.6358 X_1^2 + 3.5252 X_2^2 + 0.0838 X_3^2\| \right\}. \end{aligned}$$

where the matrix  $P_i$  for  $i=1, 2, 3$  is given by equation (4.78), and  $\epsilon_1 = 0.6$ ,  $\epsilon_2 = 0.3$ ,  $\epsilon_3 = 0.7$ .

In the simulation, the manipulator is required to move from  $\theta(0) = [-0.5, -1.2, 0.2]^T$  radians to  $\theta(\tau) = [1.0, 0.2, 1.2]^T$  radians in  $\tau = 2$  seconds while carrying a 10 Kg. load.

The graphs indicating the behaviour of the system are shown in Figures 6.5 to 6.9. The simulations show that the manipulator joints track the desired commands (position, velocity, and acceleration) with very small tracking errors. This is as expected because the controller is similar to the decentralized global

### Joint Tracking Response For 10 Kg Load

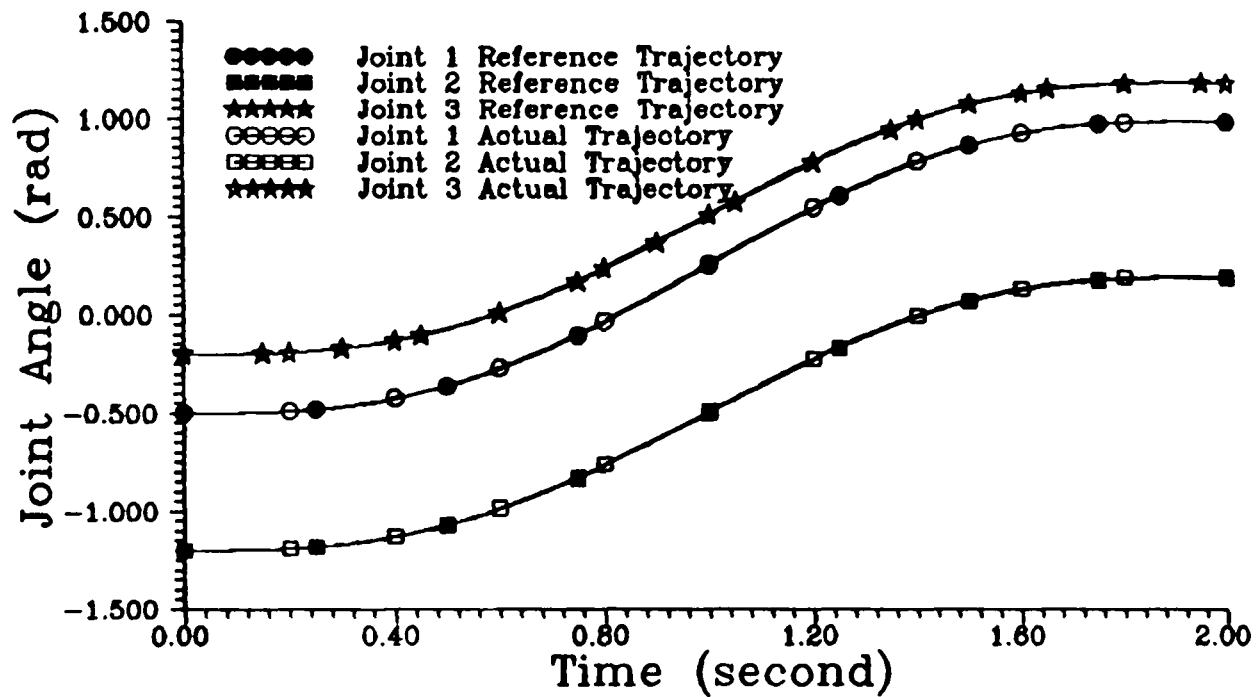


FIGURE 6.5 : Position Tracking Responses For 10 Kg. Load  
Using Hierarchical Control - Method 1

### Joint 1 Tracking Error

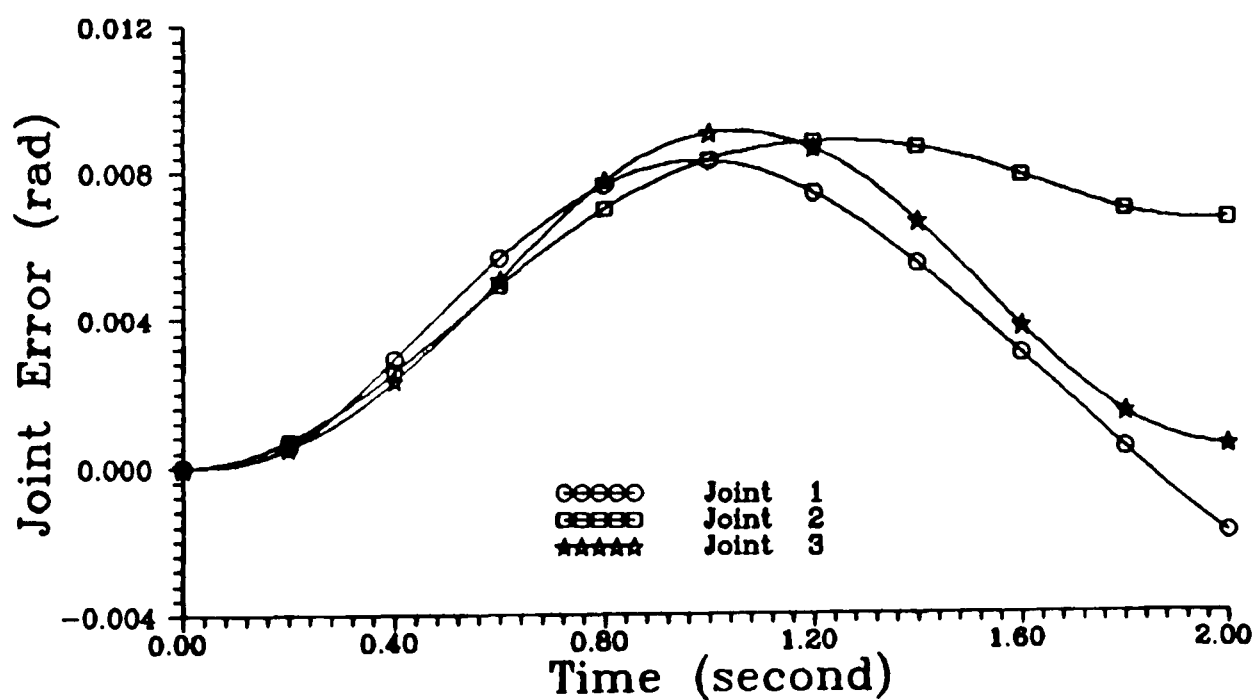


FIGURE 6.6 : Position Tracking Error For 10 Kg. Load  
Using Hierarchical Control - Method 1

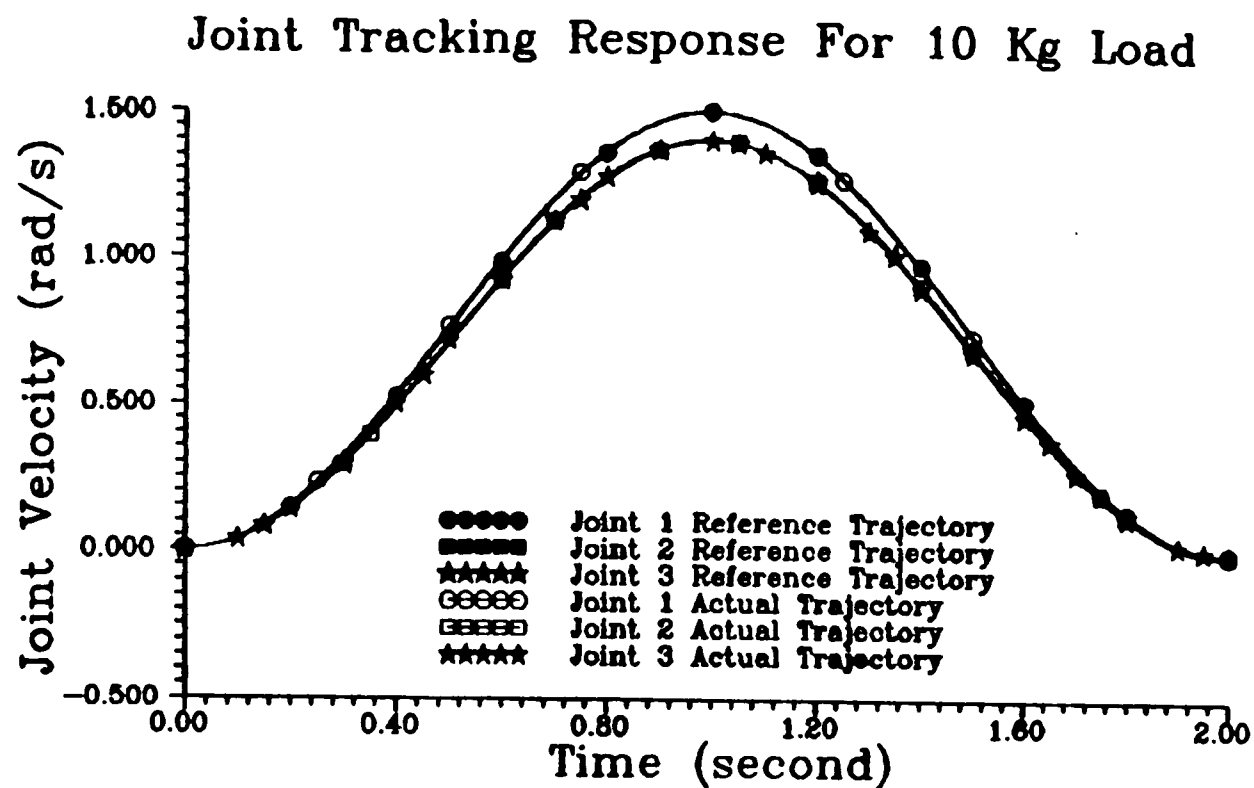


FIGURE 6.7 : Velocity Tracking Responses For 10 Kg. Load  
Using Hierarchical Control – Method 1

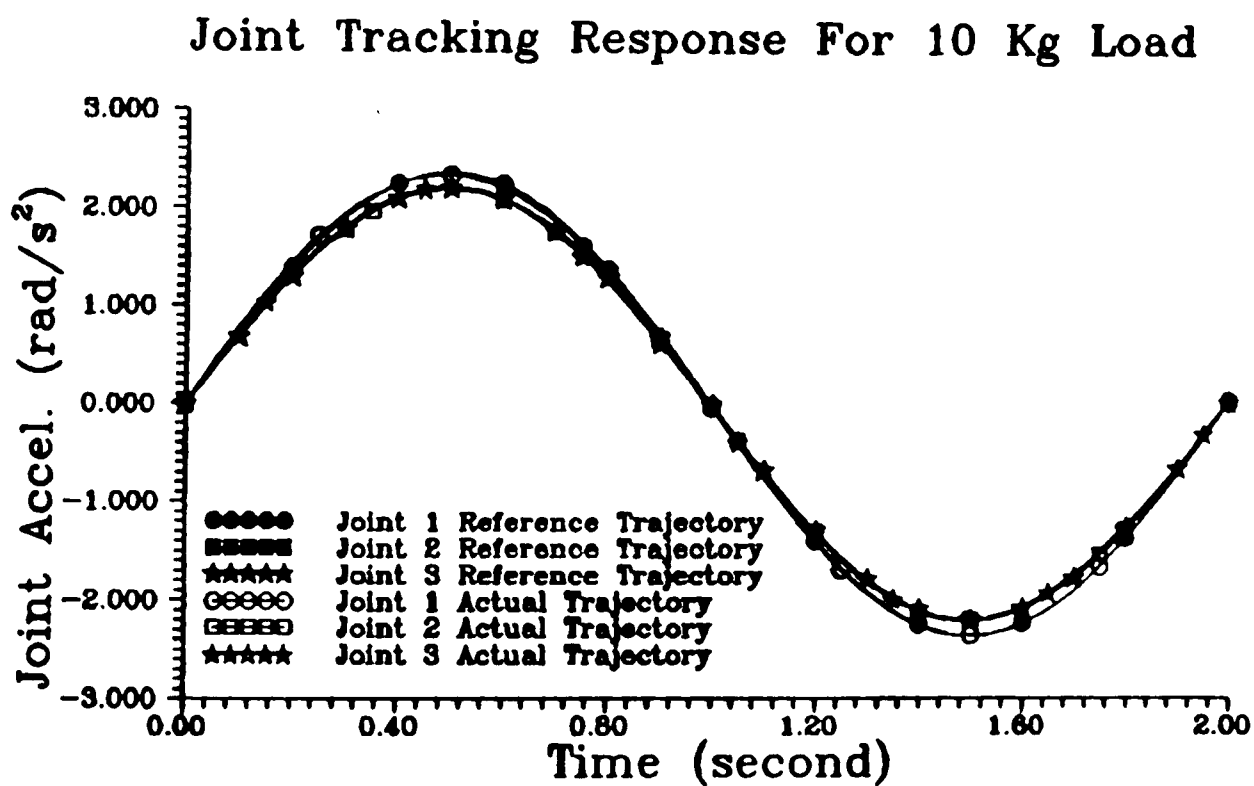


FIGURE 6.8 : Acceleration Tracking Responses For 10 Kg. Load  
Using Hierarchical Control – Method 1

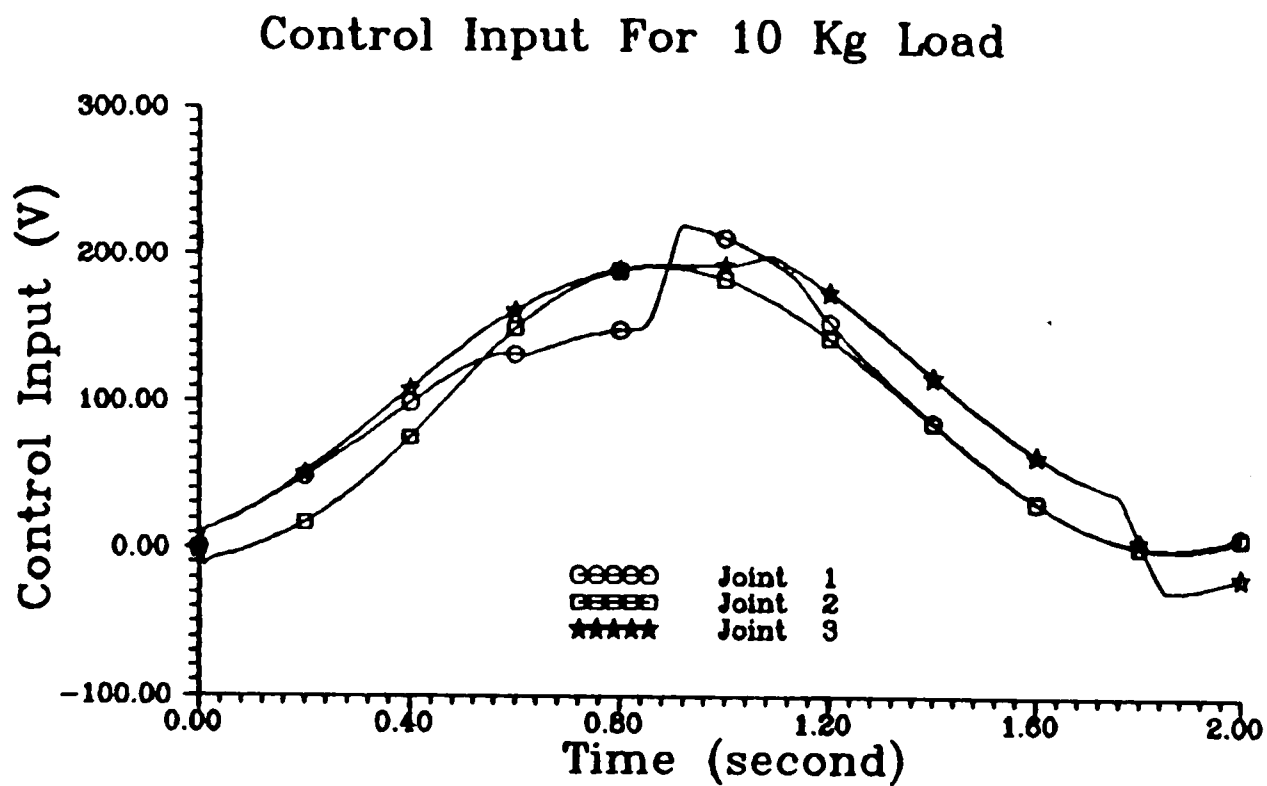


FIGURE 6.9 : Joint Control Inputs For 10 Kg. Load Using Hierarchical Control – Method 1

controller as presented in Chapter 5, except that in this case, the nonlinear part of the controller has been placed at the upper level of the control hierarchy. Thus, the controller will have the same characteristic as the decentralized global controller if the sampling interval at both of the levels are similar, and if there is no interruption in communication between the upper and the lower level.

#### 6.5.2 Implementation Of The Second Hierarchical Control Method

For this method, both levels use a controller synthesized based on a deterministic approach. For the lowest level, the decentralized control law is designed based on equations (6.30) to (6.33) :



$$\Phi_i^{L2}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_i(Z_i, \xi, t)}{\|\mu_i(Z_i, \xi, t)\|} \rho_i^{L2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| > \epsilon_i \\ -\frac{\mu_i(Z_i, \xi, t)}{\epsilon_i} \rho_i^{L2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (6.77)$$

$$\mu_i(Z_i, \xi, t) = B_i^T P_i Z_i(t) \quad (6.78)$$

$$\begin{aligned} \rho_1^{L2}(Z_1, \xi, t) &= 1.0339 \left\{ \|4.1174 X_2^1 + 0.5271 X_3^1\| + \|0.0328 \tilde{U}_1\| \right\} \\ \rho_2^{L2}(Z_2, \xi, t) &= 1.0256 \left\{ \|28.1056 X_1^2 + 3.9371 X_2^2 + 0.1489 X_3^2\| + \right. \\ &\quad \left. \|0.0249 \tilde{U}_2\| \right\} \end{aligned} \quad (6.79)$$

$$\begin{aligned} \rho_3^{L2}(Z_3, \xi, t) &= 1.0003 \left\{ \|12.5947 X_1^3 + 0.3616 X_2^3 + 2.6029 \times 10^{-3} X_3^3\| + \right. \\ &\quad \left. \|2.6154 \times 10^{-4} \tilde{U}_3\| \right\}, \end{aligned}$$

$$\epsilon_1 = 0.6, \quad \epsilon_2 = 0.3, \quad \epsilon_3 = 0.7,$$

and the same feedback gains  $K_i$ , the continuous functions  $\Omega_i(t)$ , and the matrix  $P_i$  for  $i = 1, 2, 3$  as previously used for the first hierarchical method are utilized.

Based on equations (6.45) to (6.47), the coordinator for the upper level is obtained as follows :

$$\Phi_i^{U2}(Z_i, \xi, t) = \begin{cases} -\frac{\mu_i(Z_i, \xi, t)}{\|\mu_i(Z_i, \xi, t)\|} \rho_i^{U2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| > \epsilon_i \\ -\frac{\mu_i(Z_i, \xi, t)}{\epsilon_i} \rho_i^{U2}(Z_i, \xi, t) & \text{if } \|\mu_i(Z_i, \xi, t)\| \leq \epsilon_i \end{cases} \quad (6.80)$$

$$\rho_1^{U2}(Z_1, \xi, t) = 1.0339 \left\{ \|2.2809 X_2^2 + 0.1964 X_3^2\| + \right.$$

$$\| 1.3024 X_2^3 + 0.1153 X_3^3 \| \}$$

$$\rho_2^{U2}(Z_2, \xi, t) = 1.0256 \left\{ \| 0.9637 X_2^1 + 0.1687 X_3^1 \| + \right. \\ \left. \| 0.2816 X_1^3 + 3.9323 X_2^3 + 0.2094 X_3^3 \| \right\} \quad (6.81)$$

$$\rho_3^{U2}(Z_3, \xi, t) = 1.0003 \left\{ \| 0.579 X_2^1 + 0.1016 X_3^1 \| + \right. \\ \left. \| 0.6358 X_1^2 + 3.5252 X_2^2 + 0.0838 X_3^2 \| \right\},$$

and  $\epsilon_i$  is similar as the first level.

The simulation results obtained for this particular control indicate that the controller can cope with the nonlinear and highly coupled robot manipulator satisfactorily (Figures 6.10 to 6.14).

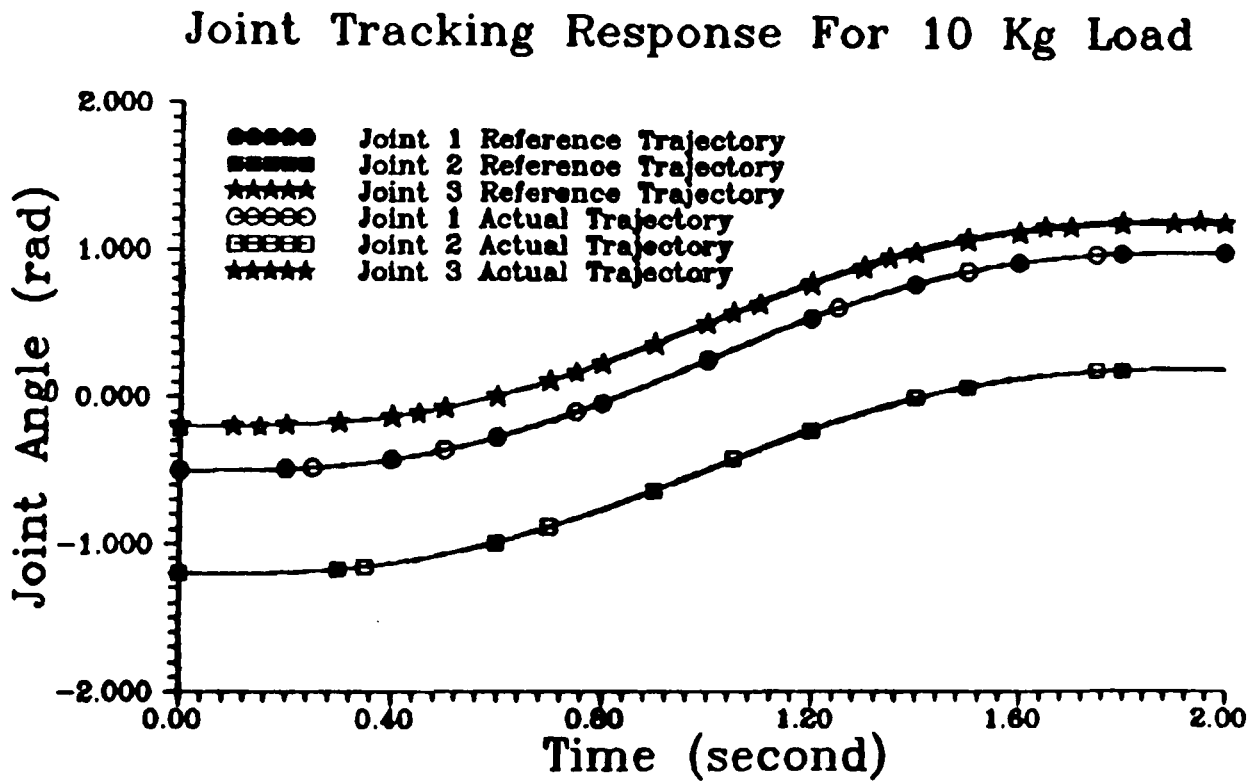


FIGURE 6.10 : Position Tracking Responses For 10 Kg. Load  
Using Hierarchical Control – Method 2

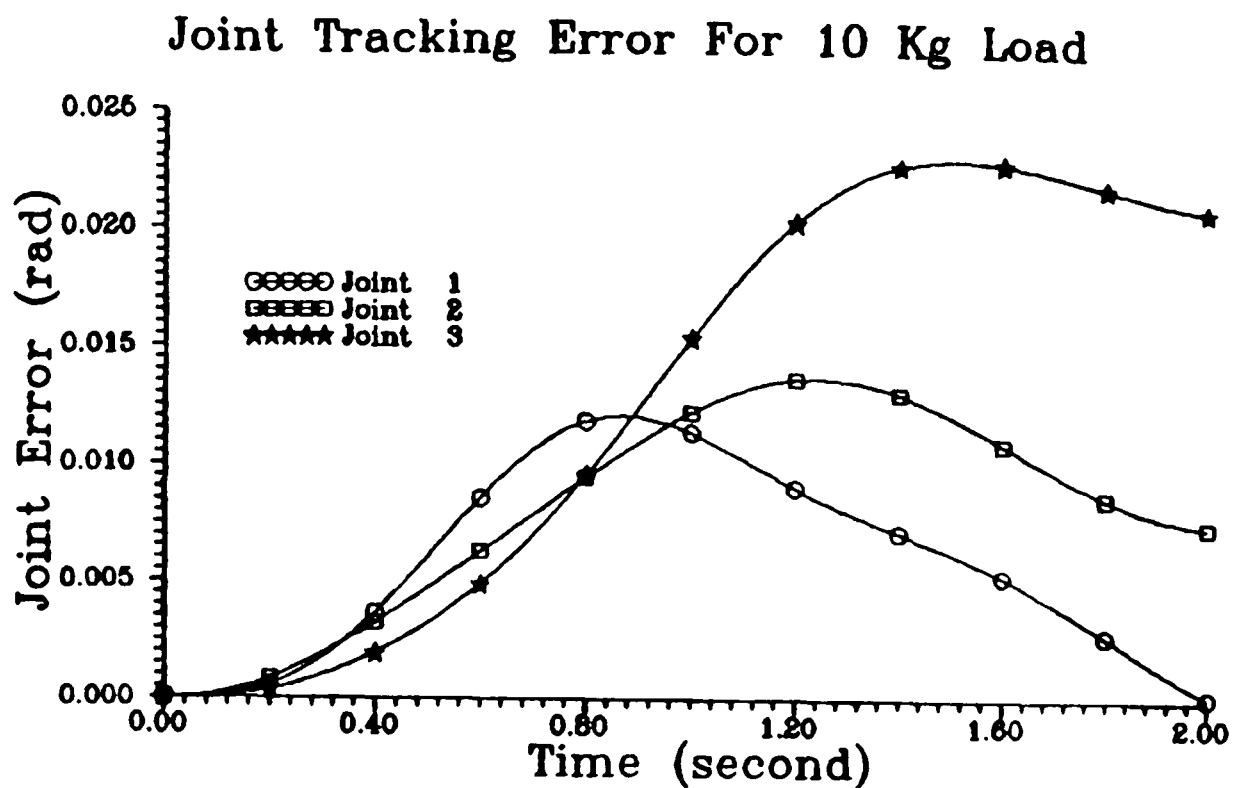


FIGURE 6.11 : Position Tracking Error For 10 Kg. Load  
Using Hierarchical Control – Method 2

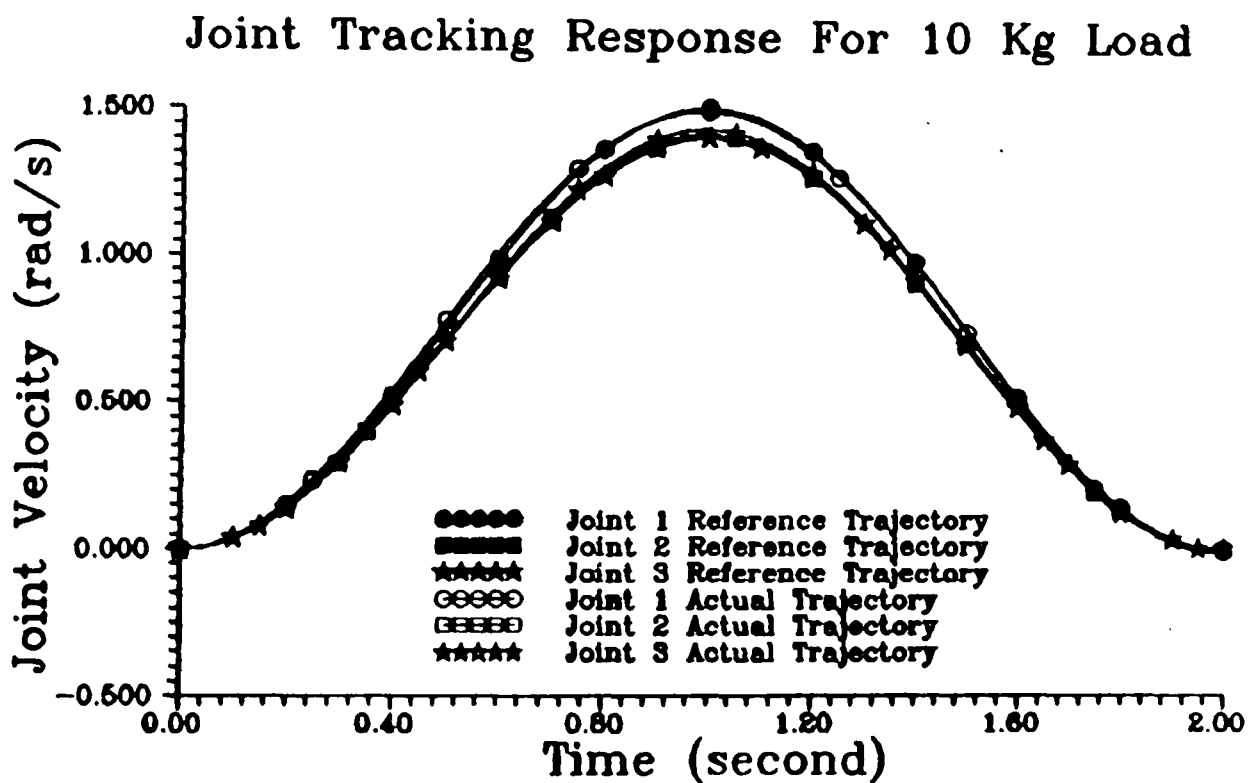


FIGURE 6.12 : Velocity Tracking Responses For 10 Kg. Load  
Using Hierarchical Control – Method 2

Joint Tracking Response For 10 Kg Load

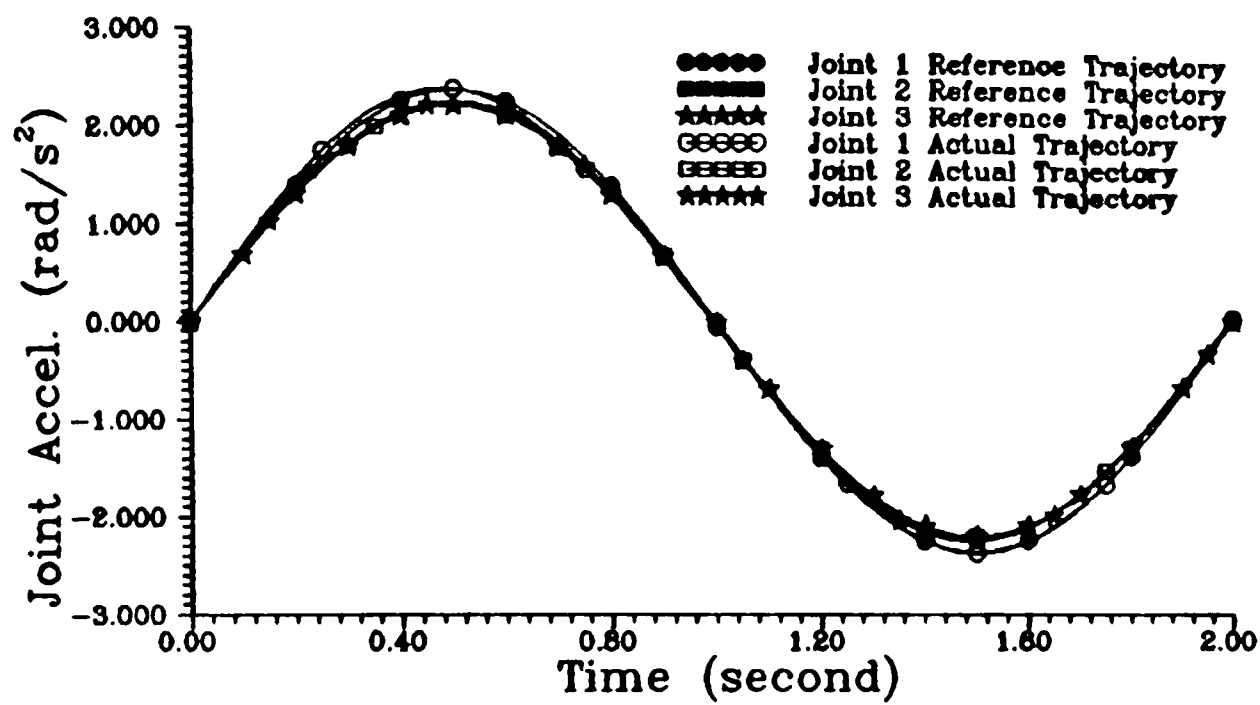


FIGURE 6.13 : Acceleration Tracking Responses For 10 Kg. Load Using Hierarchical Control – Method 2

Control Input For 10 Kg Load

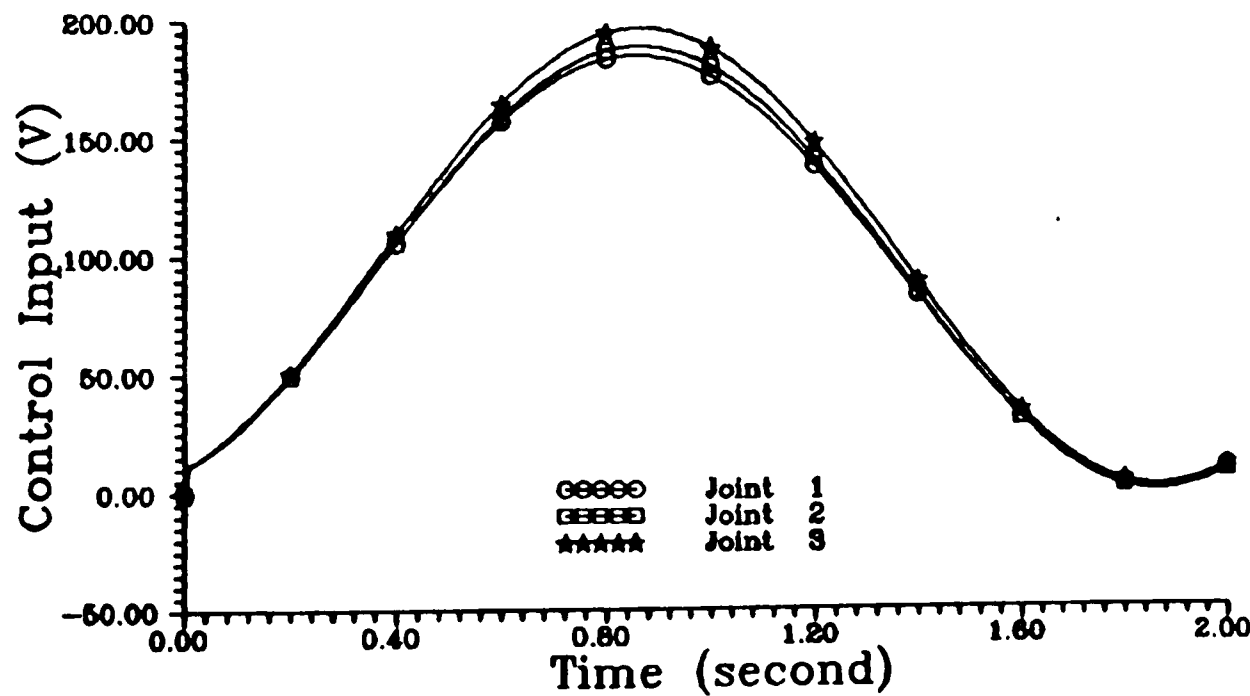


FIGURE 6.14 : Joint Control Inputs For 10 Kg. Load Using Hierarchical Control – Method 2

By comparing Figures 6.5 and 6.9 to Figures 6.10 and 6.14, respectively, it can be observed that under the same conditions and controller parameters used, the first method produced slightly better tracking accuracy than the second hierarchical control strategy. However, the second control method uses less input energy than the first method. Thus, there is a trade off between the two hierarchical control methods in term of tracking accuracy and input energy required.

### 6.5.3 Effect Of Load Variation

Figures 6.15, 6.16, and 6.17 illustrate the joint 1, joint 2 , and joint 3 tracking responses, respectively, of the robot manipulator using the first hierarchical control method under various loading conditions. The manipulator is required to track the desired trajectory while moving without load, with a 10 Kg. load, and finally, with a 20 Kg. load. Figures 6.18 and 6.19 show the control inputs for no load and 20 Kg. load, respectively.

The performance of the second hierarchical control strategy under different loading conditions is as depicted in Figures 6.20 through 6.24. As expected, the tracking errors between the various loading conditions are small for both methods. The control inputs are smooth, continuous, and there is no significant rise in the control input demand when the load is increase to 20 Kg from 0 Kg load. Thus, for both methods, the same controller can force the robot manipulator under the various loading conditions with relatively small tracking errors. And the tracking error for the manipulator carrying a load within the specified range (0 – 20 Kg.) will fall within the 0 and 20 Kg. load curves in the graphs. If the difference between the tracking error for 20 Kg load and that of without load is acceptable for the designer, then, the controllers are robust to the load variations present in the system.

However, comparing the two control strategies under the load variations, the first method is better than the second method in terms of tracking precision.

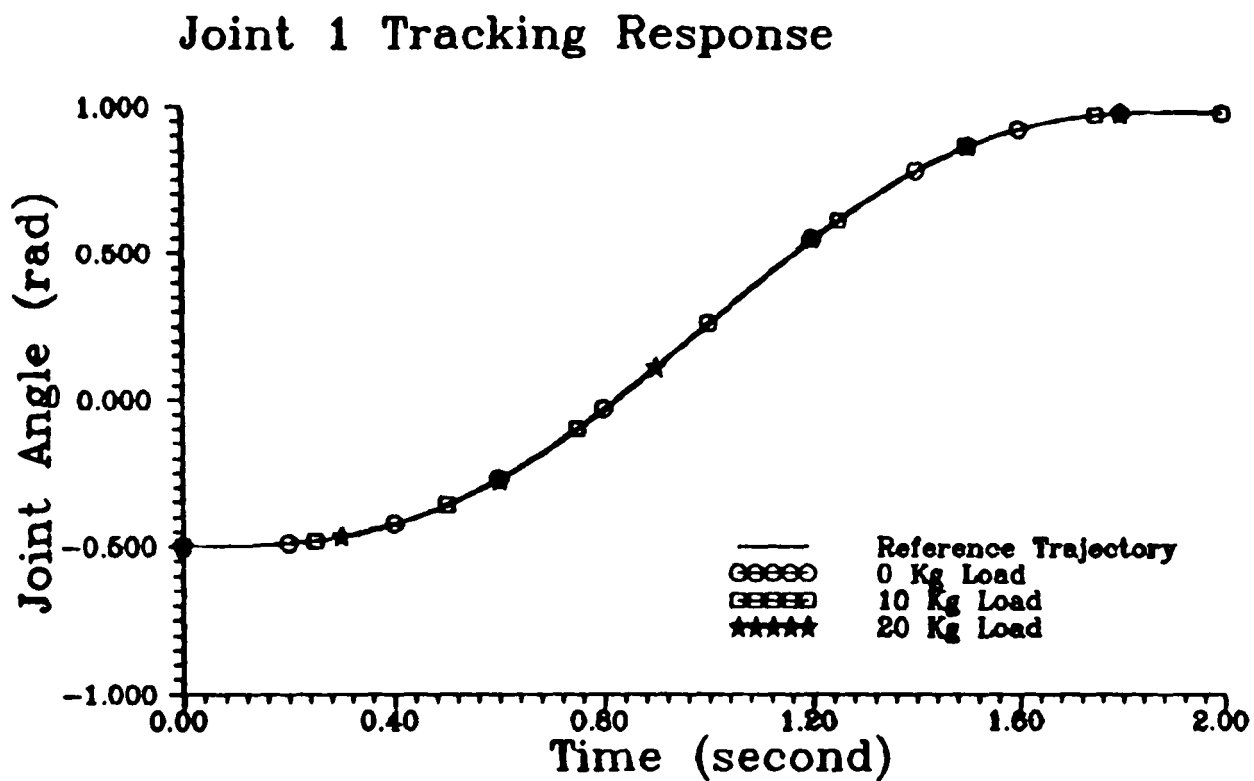


FIGURE 6.15 a : Joint 1 Tracking Response Under Different Load  
Using Hierarchical Control – Method 1

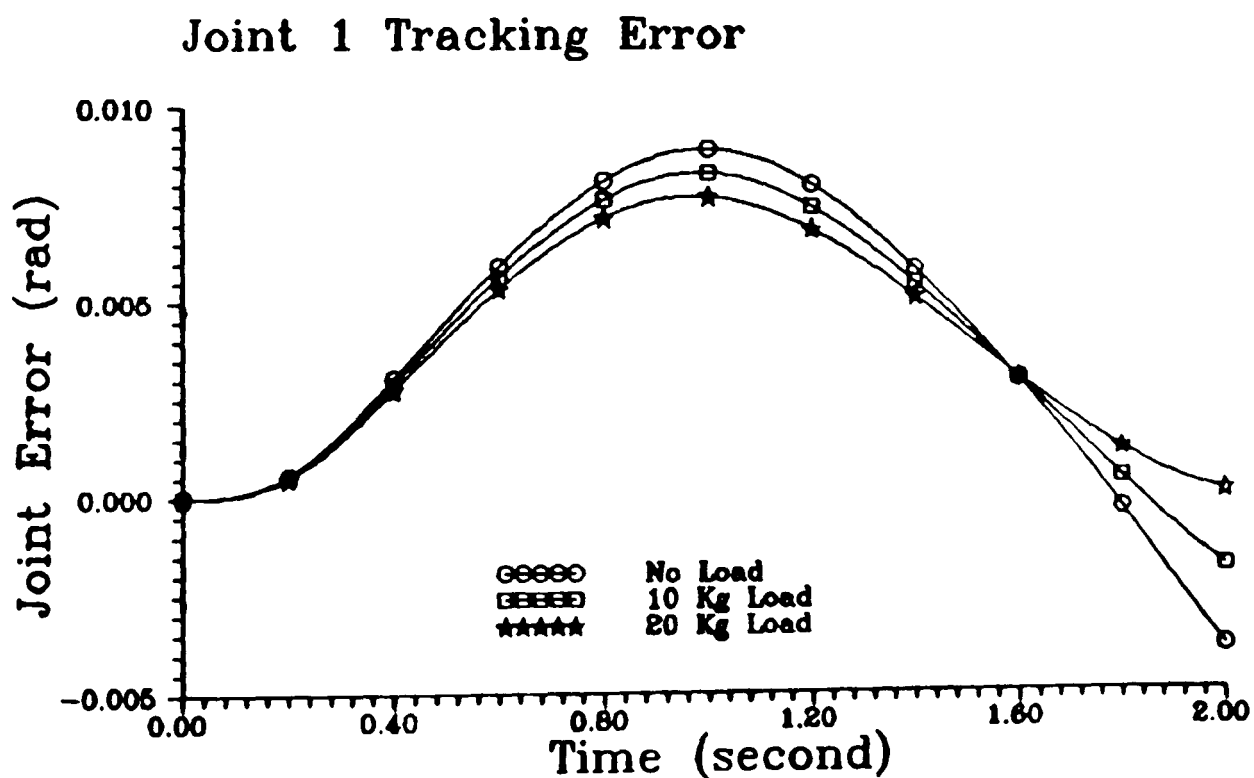


FIGURE 6.15 b : Joint 1 Tracking Error Under Different Load  
Using Hierarchical Control – Method 1

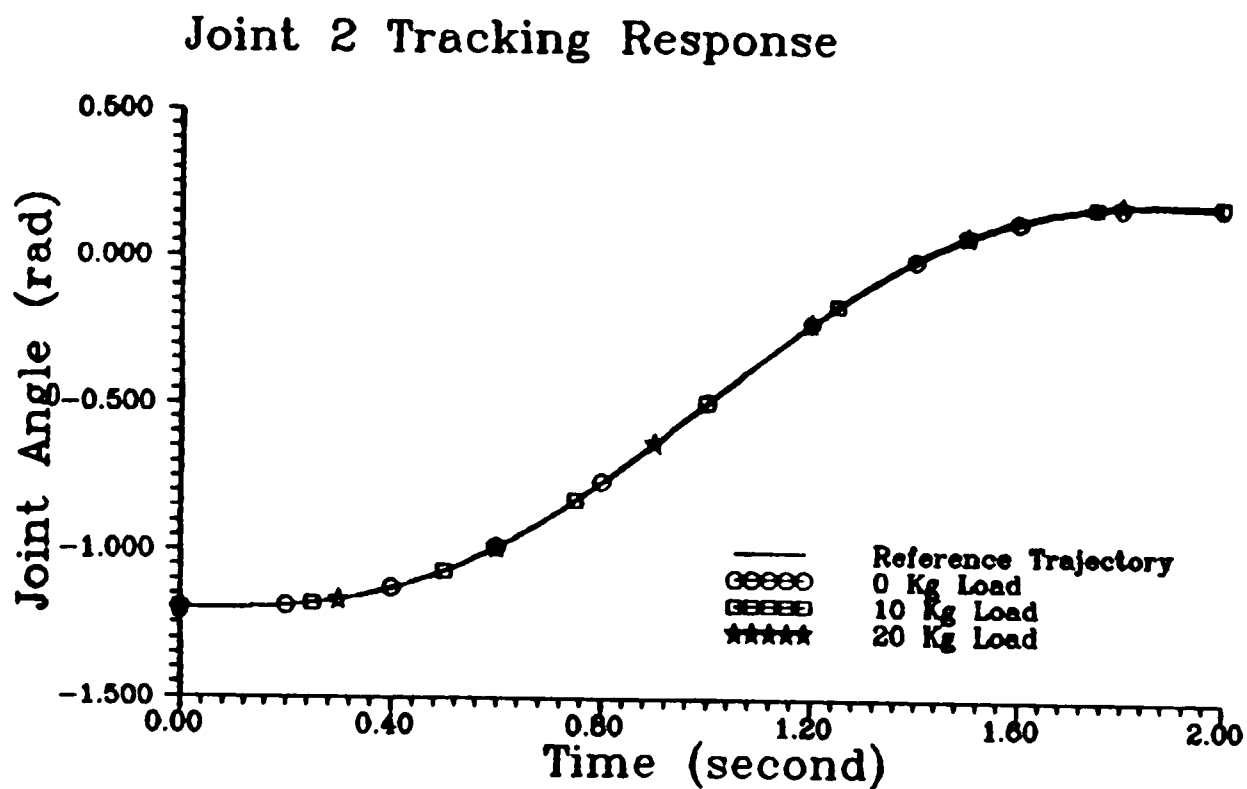


FIGURE 6.16 a : Joint 2 Tracking Response Under Different Load Using Hierarchical Control – Method 1

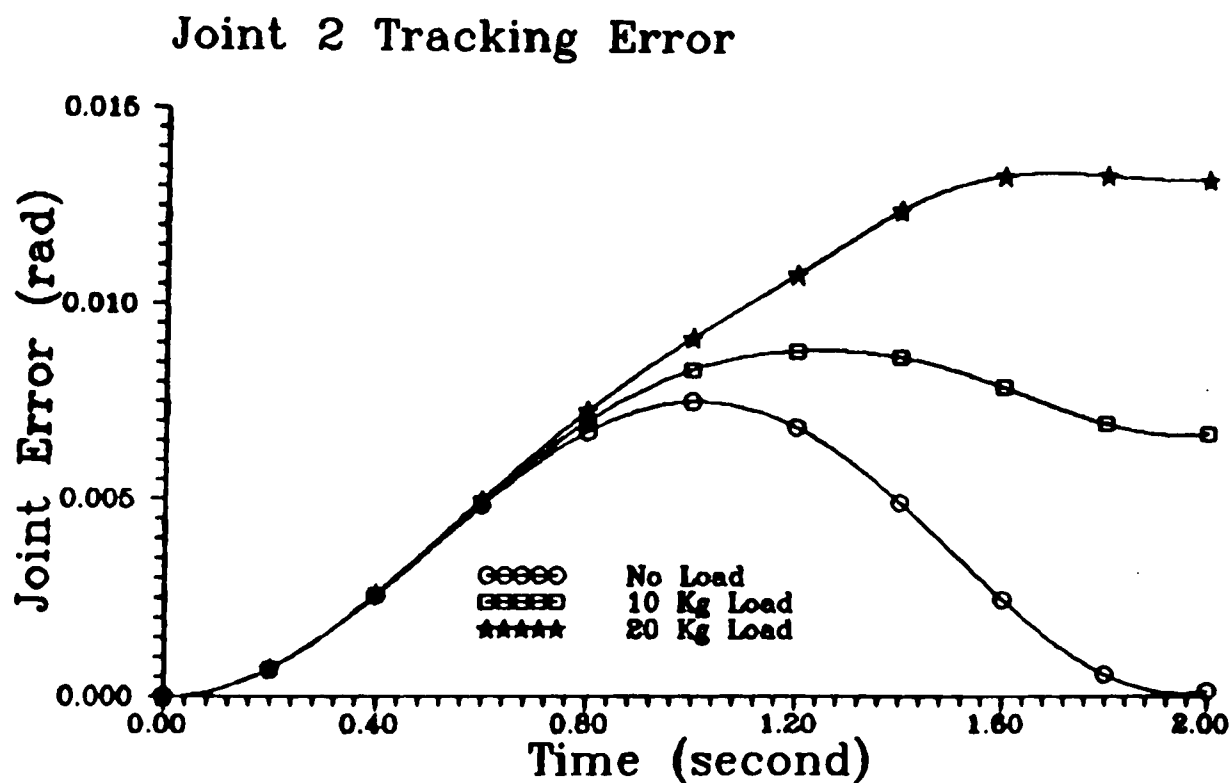


FIGURE 6.16 b : Joint 2 Tracking Error Under Different Load Using Hierarchical Control – Method 1

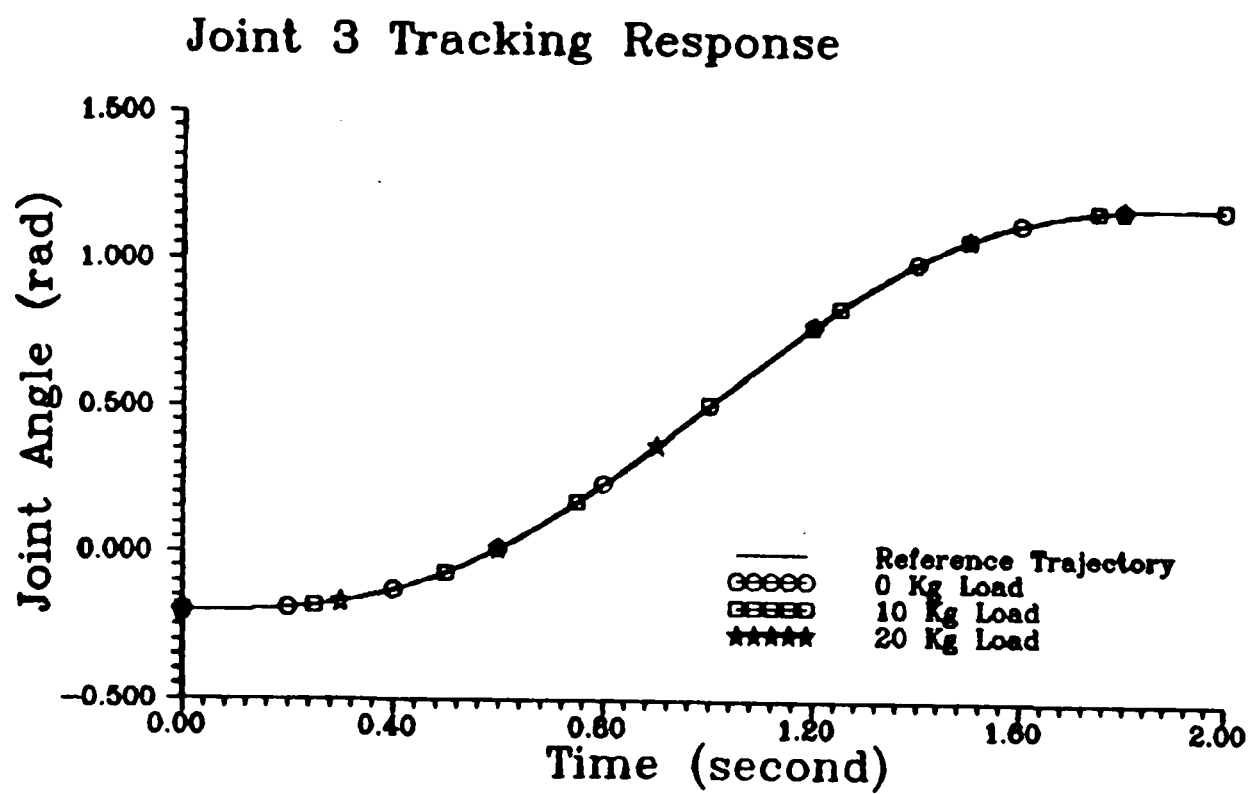


FIGURE 6.17 a : Joint 3 Tracking Response Under Different Load  
Using Hierarchical Control – Method 1

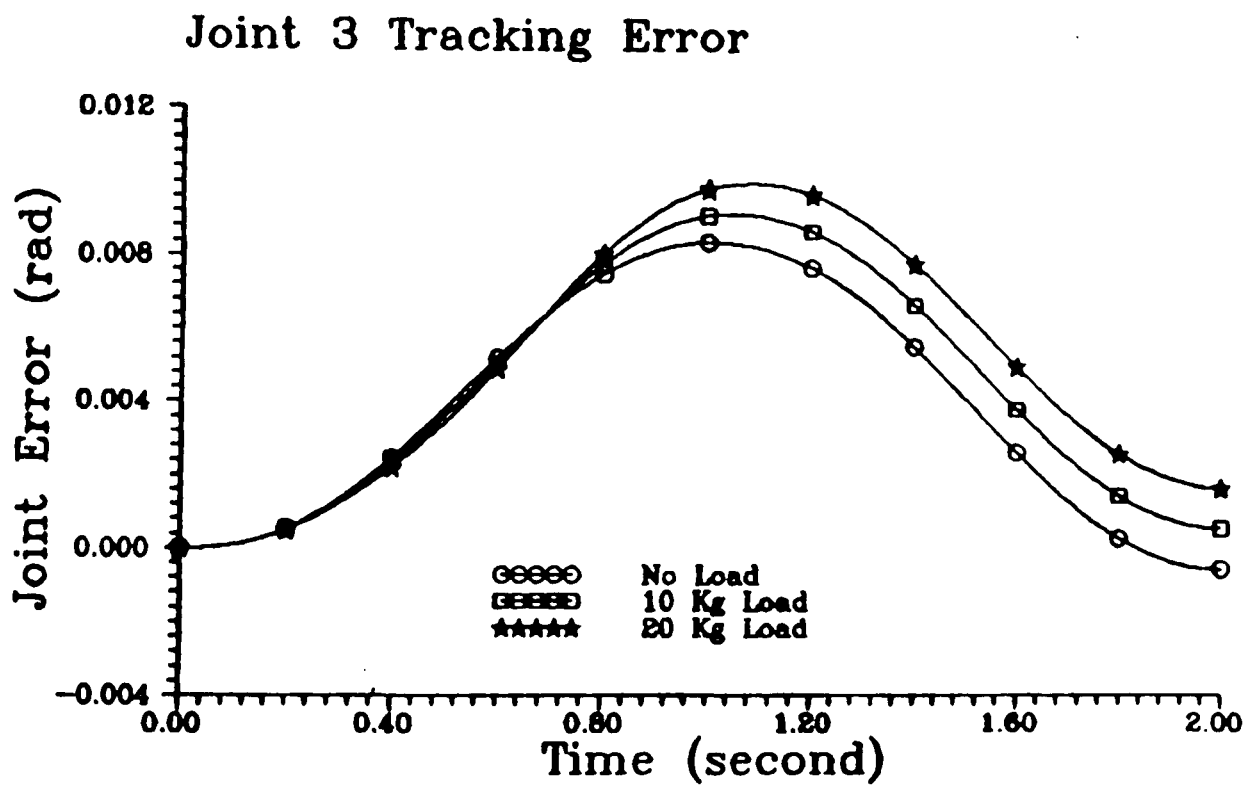


FIGURE 6.17 b : Joint 3 Tracking Error Under Different Load  
Using Hierarchical Control – Method 1



### Control Input For 0 Kg Load

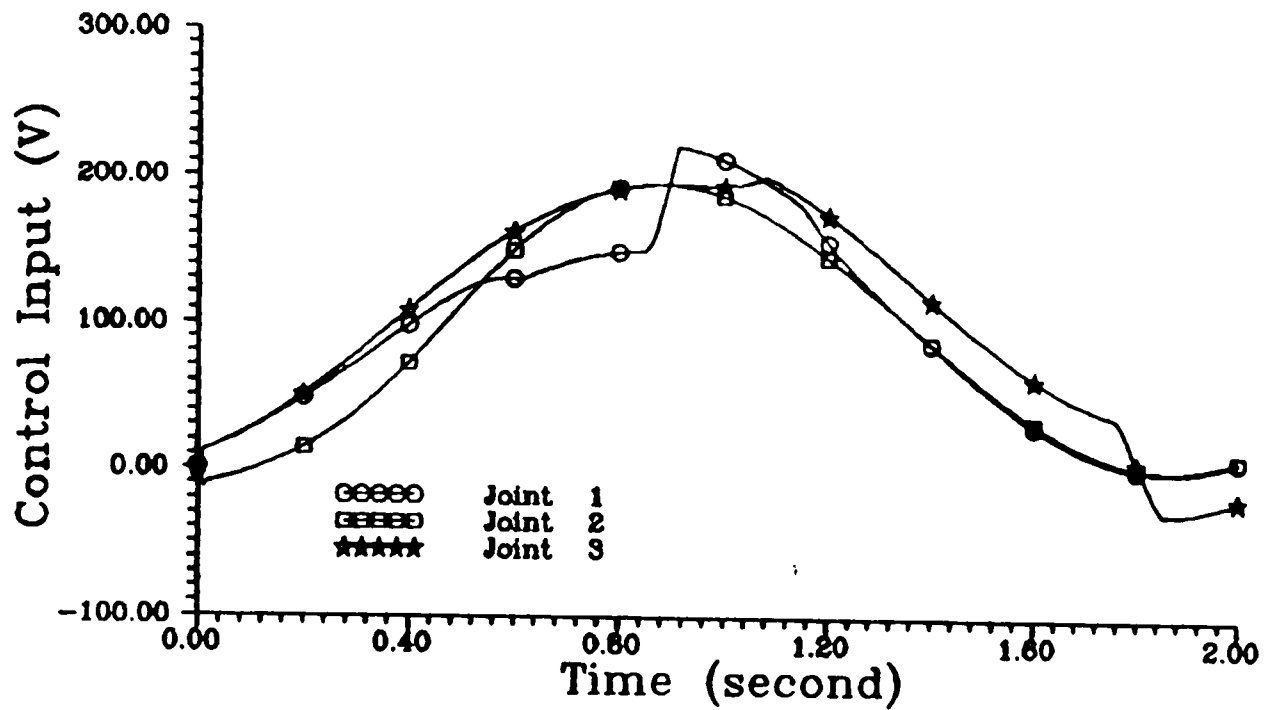


FIGURE 6.18 Control Inputs For 0 Kg. Load Using Hierarchical Control – Method 1

### Control Input For 20 Kg Load

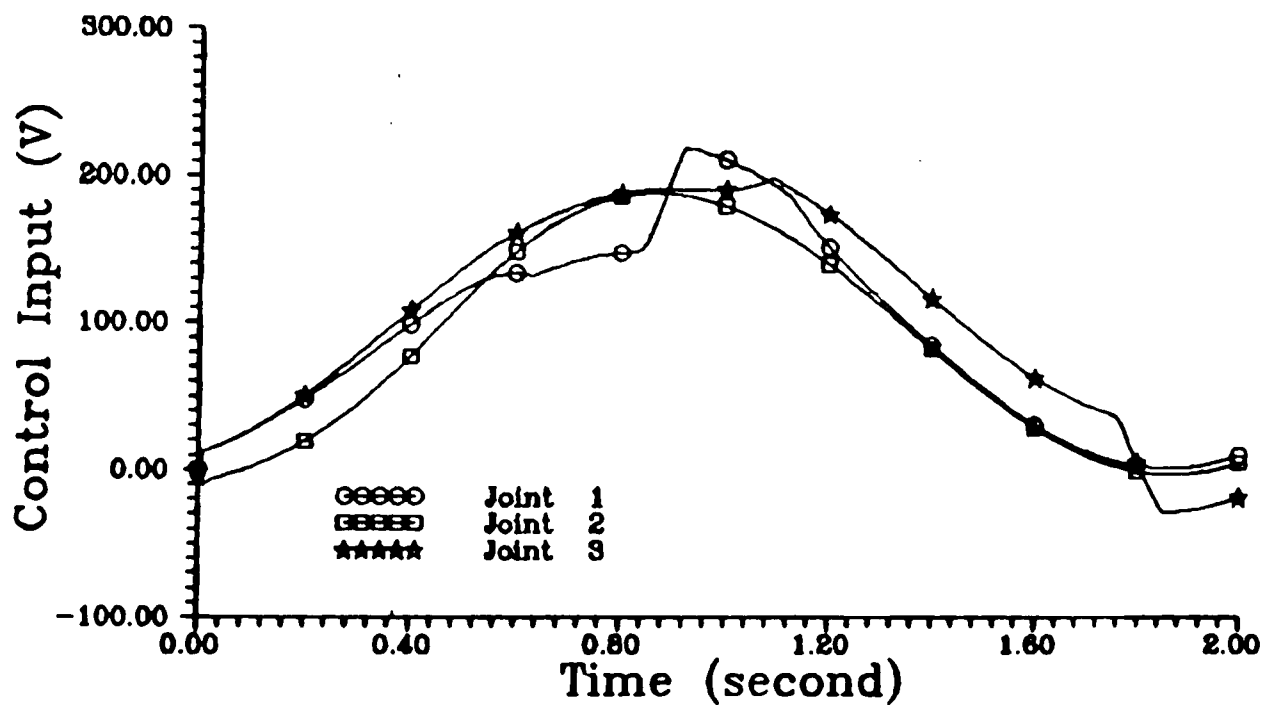


FIGURE 6.19 Control Inputs For 20 Kg. Load Using Hierarchical Control – Method 1

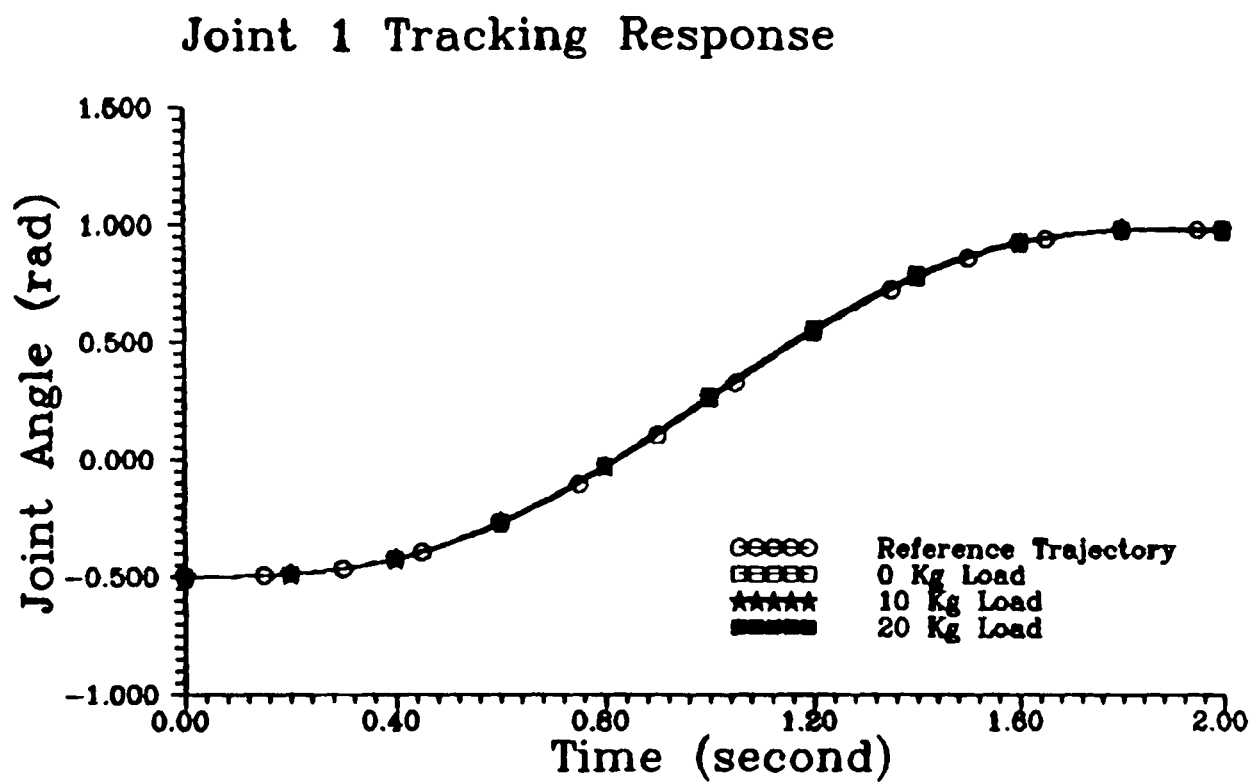


FIGURE 6.20 a : Joint 1 Tracking Response Under Different Load Using Hierarchical Control – Method 2

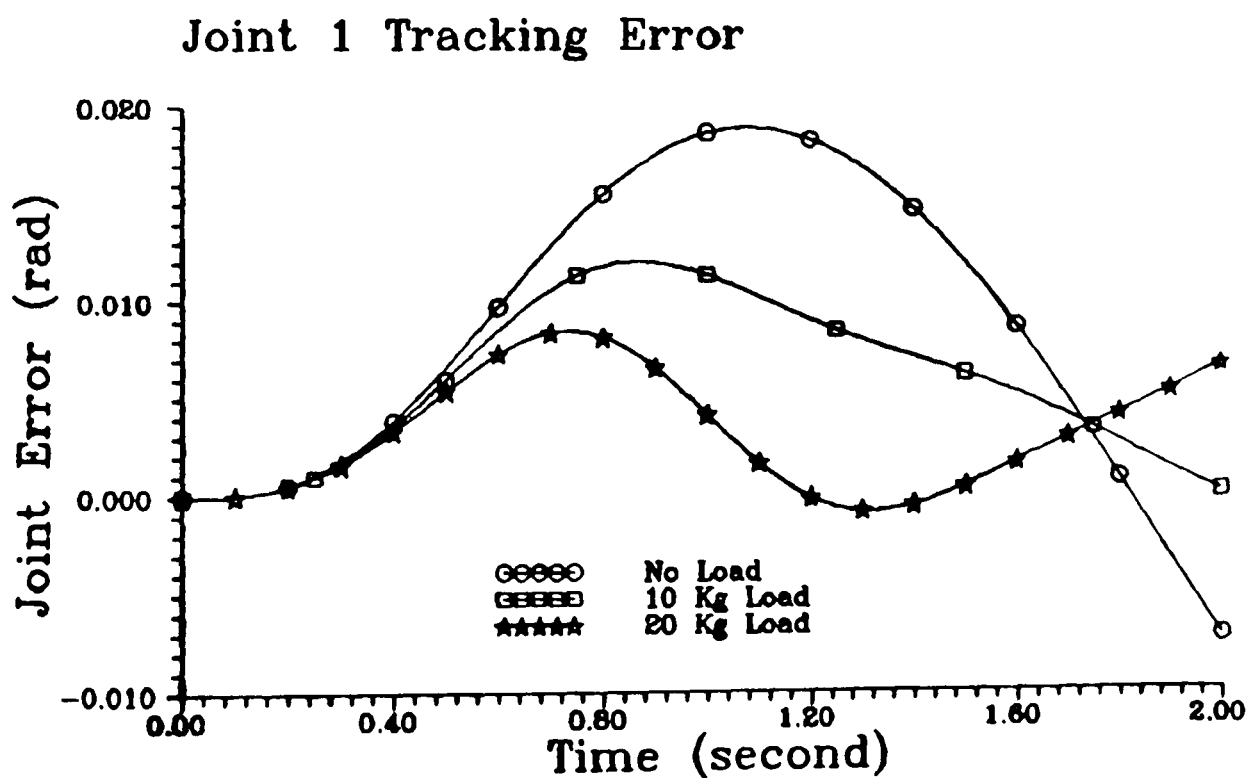


FIGURE 6.20 b : Joint 1 Tracking Error Under Different Load Using Hierarchical Control – Method 2

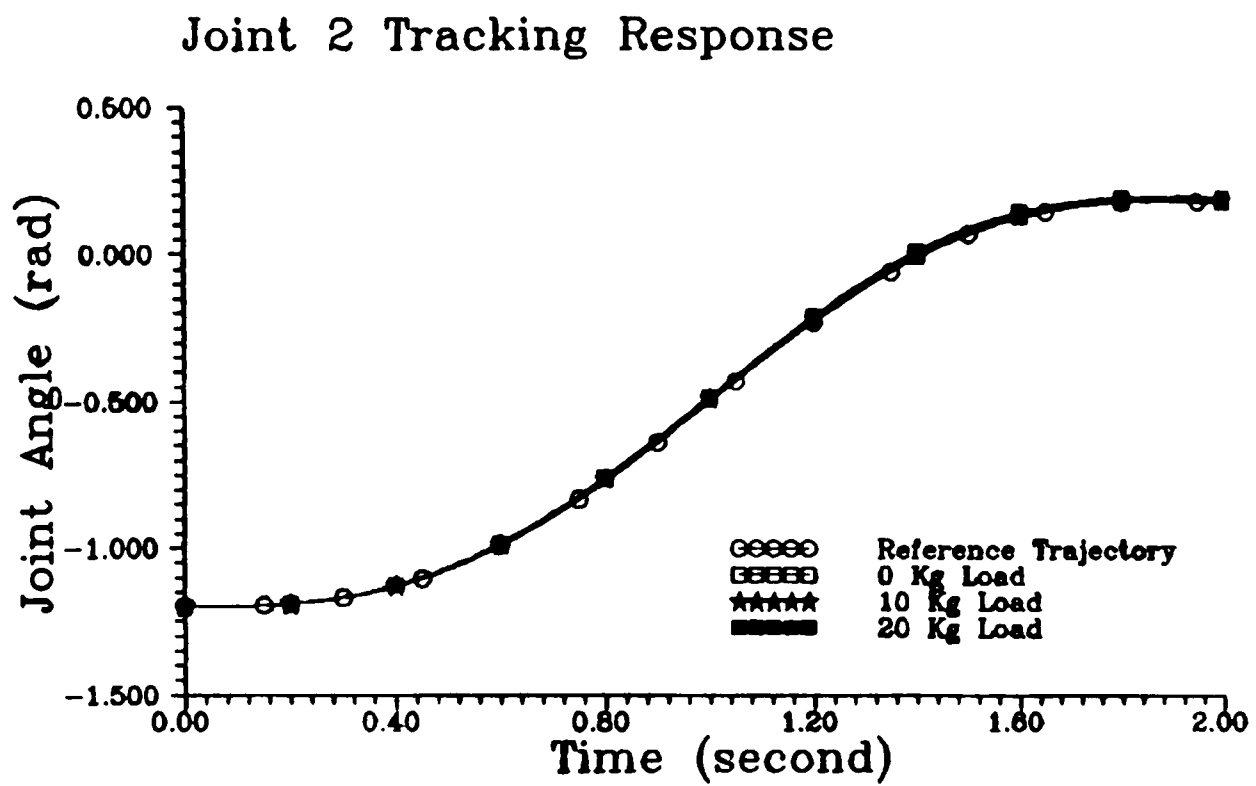


FIGURE 6.21 a : Joint 2 Tracking Response Under Different Load Using Hierarchical Control – Method 2

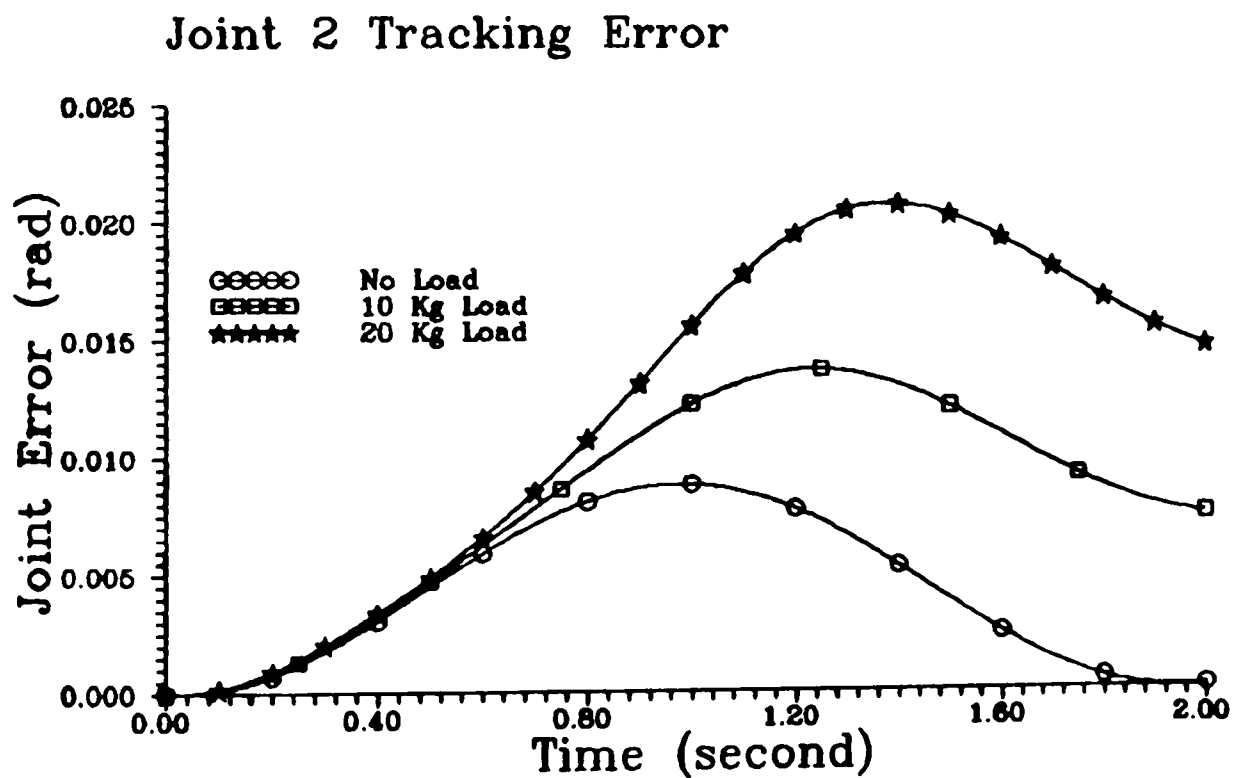


FIGURE 6.21 b : Joint 2 Tracking Error Under Different Load Using Hierarchical Control – Method 2

### Joint 3 Tracking Response

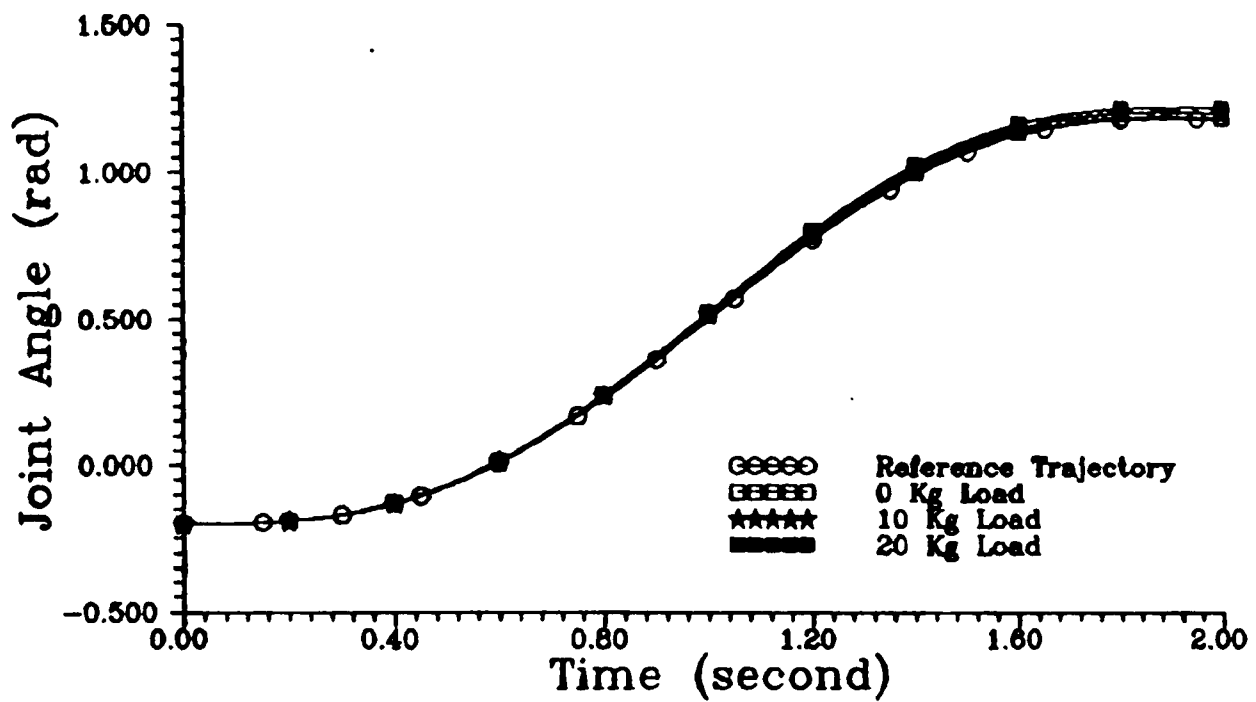


FIGURE 6.22 a : Joint 3 Tracking Response Under Different Load Using Hierarchical Control – Method 2

### Joint 3 Tracking Error

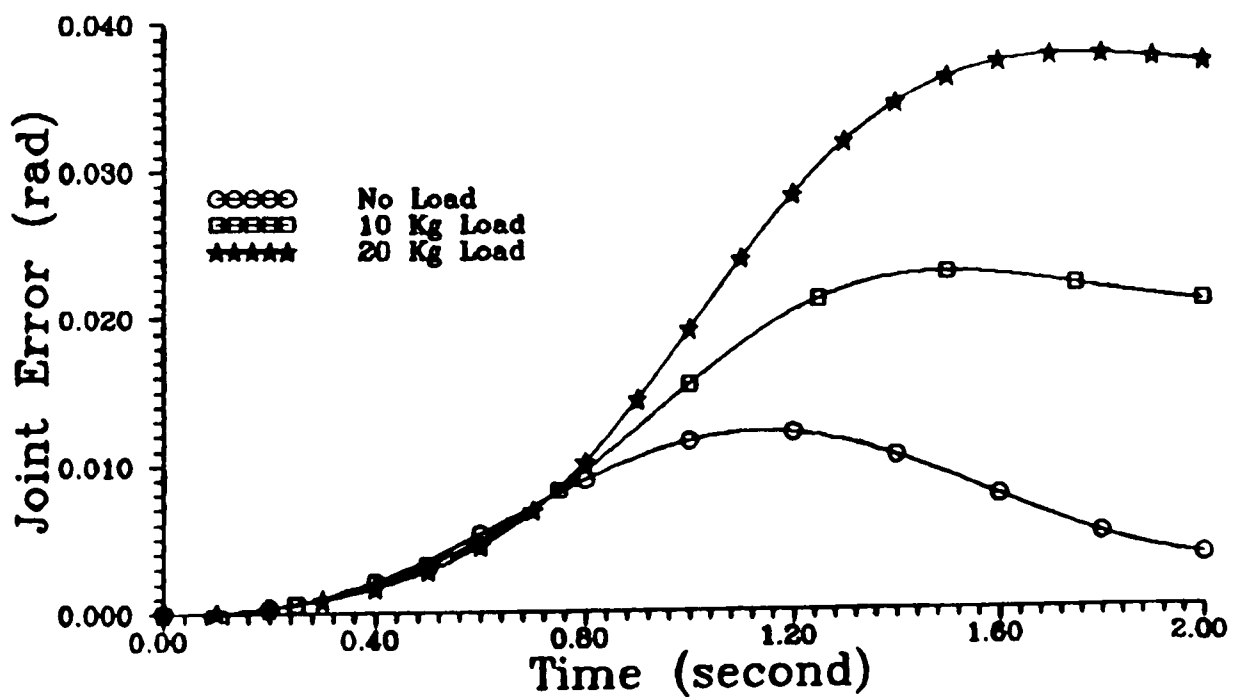


FIGURE 6.22 b : Joint 3 Tracking Error Under Different Load Using Hierarchical Control – Method 2

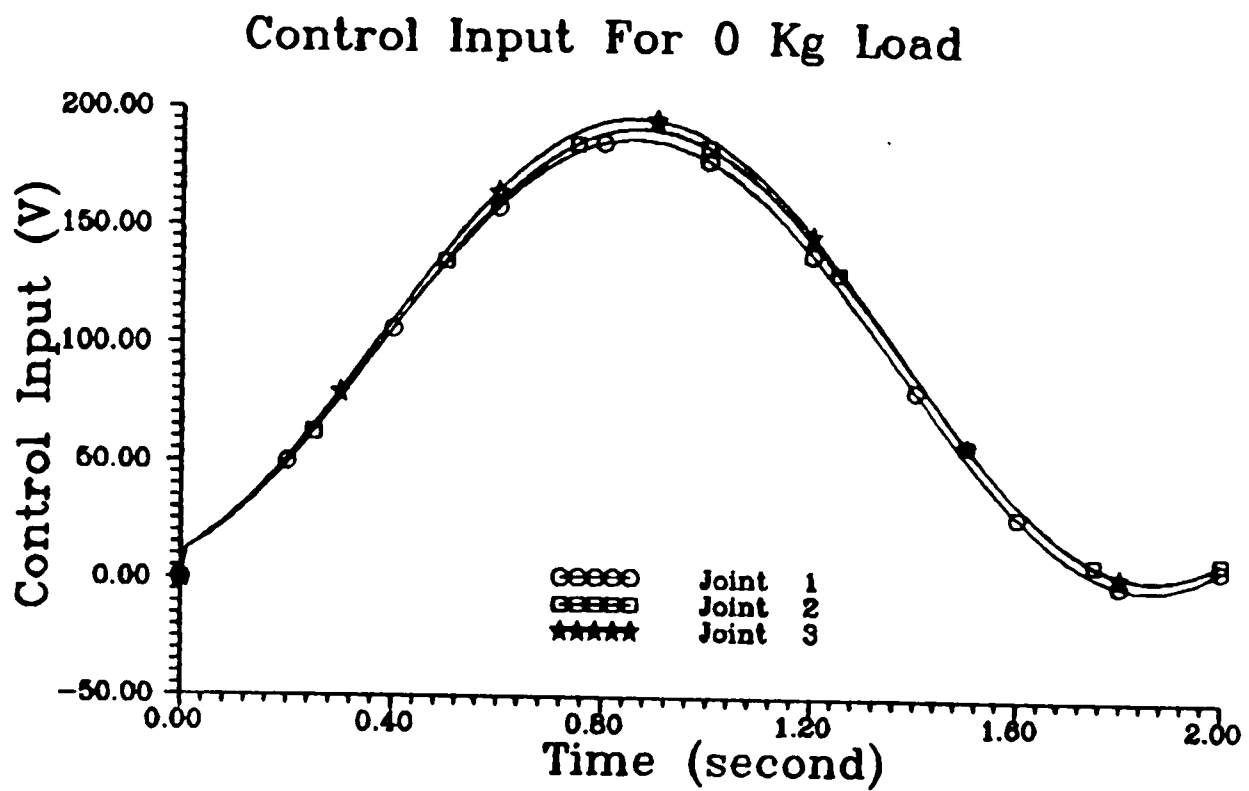


FIGURE 6.23 Control Inputs For 0 Kg. Load Using Hierarchical Control – Method 2

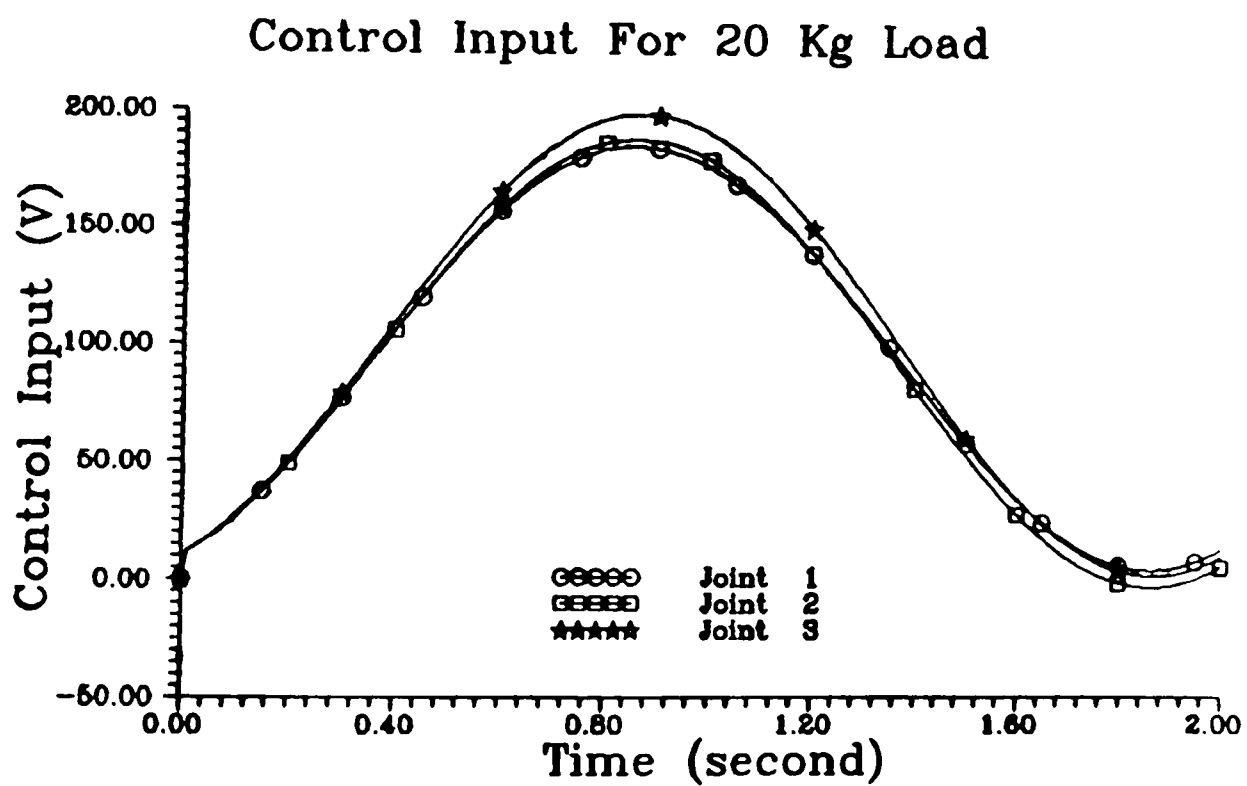


FIGURE 6.24 Control Inputs For 20 Kg. Load Using Hierarchical Control – Method 2

#### 6.5.4 Effect Of Structural Perturbation

In applying a multi-level control concept to control a complex system, there arises the possibility of structural perturbations due to the interruption of the different signal lines or the communication links within the overall system. This may tend to modify the data transmission network, and may cause information losses and, hence, may effect the stability of the global system. In this subsection, a simulation study is conducted to evaluate the performance of the two-level hierarchical controllers under various structural perturbations between the coordinator (upper level) and the subsystems (lower level). The following cases have been considered :

- Case 1 : The communication link between the coordinator and subsystem 1 is cut.
- Case 2 : The communication link between the coordinator and subsystem 2 is cut.
- Case 3 : The communication link between the coordinator and subsystem 3 is cut.
- Case 4 : All the communication links between the coordinator and the subsystems are cut. In this case, the subsystems are completely uncoordinated. Each subsystem is controlled by its own local controller only.

Figures 6.25, 6.26, and 6.27 illustrate the tracking errors for joint 1, joint 2, and joint 3, respectively, for each case for the first hierarchical control method. The simulation results for the robot manipulator using the second hierarchical control method under structural perturbation are as depicted in Figures 6.28, 6.29, and 6.30, for joint 1, joint 2, and joint 3, respectively. Also shown in all these figures is the joint tracking error for the case when there is no structural perturbation (complete hierarchy).

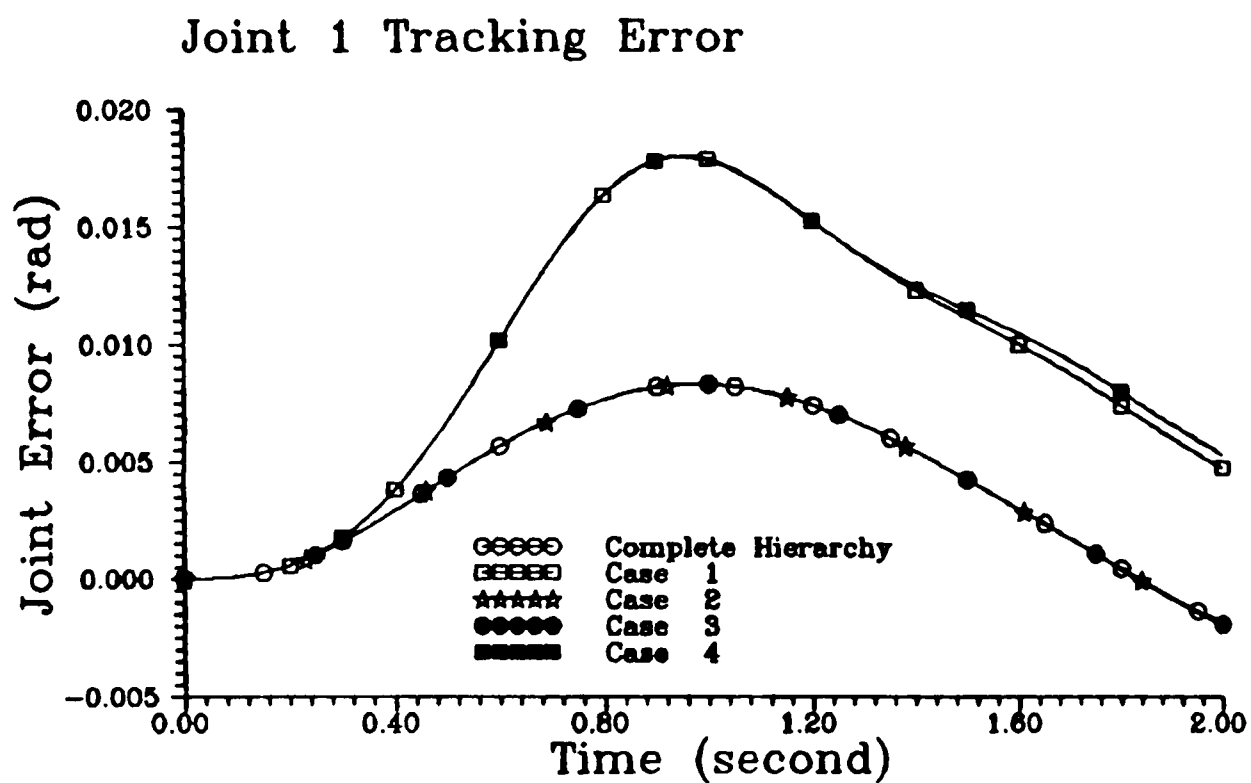


FIGURE 6.25 Joint 1 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 1

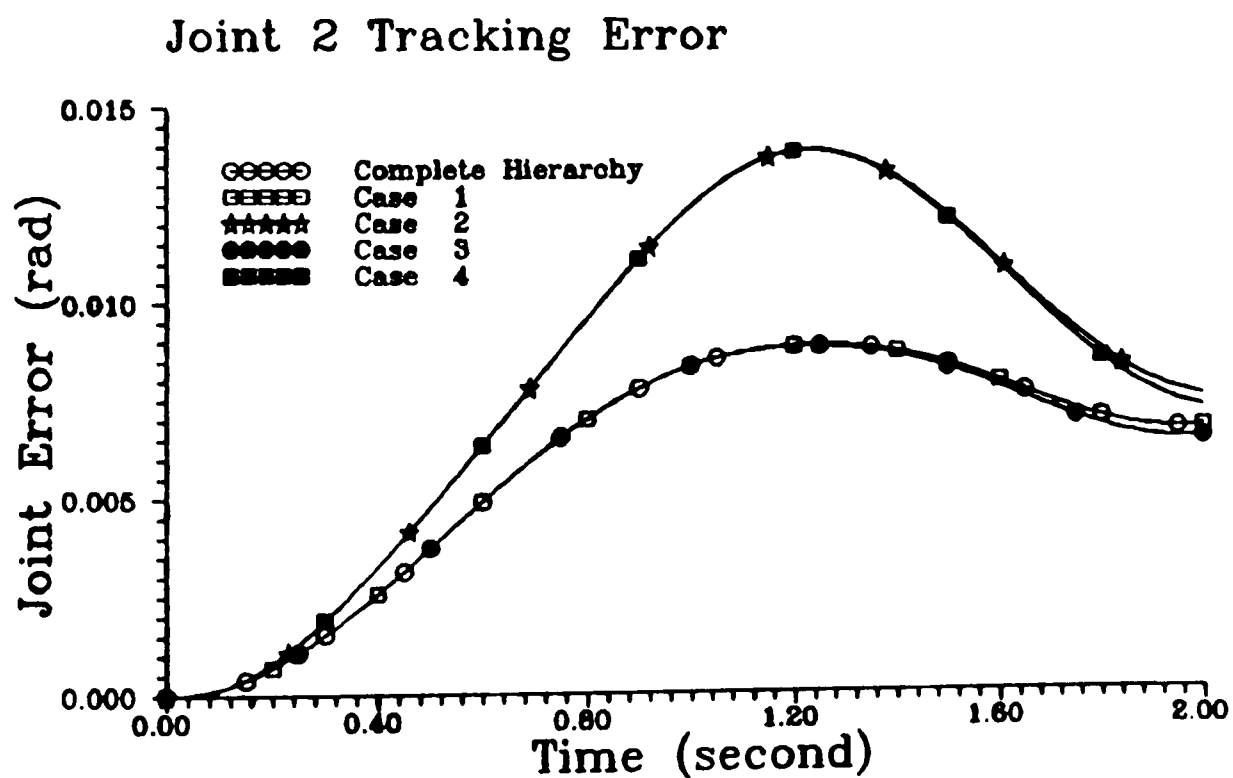


FIGURE 6.26 Joint 2 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 1

### Joint 3 Tracking Error

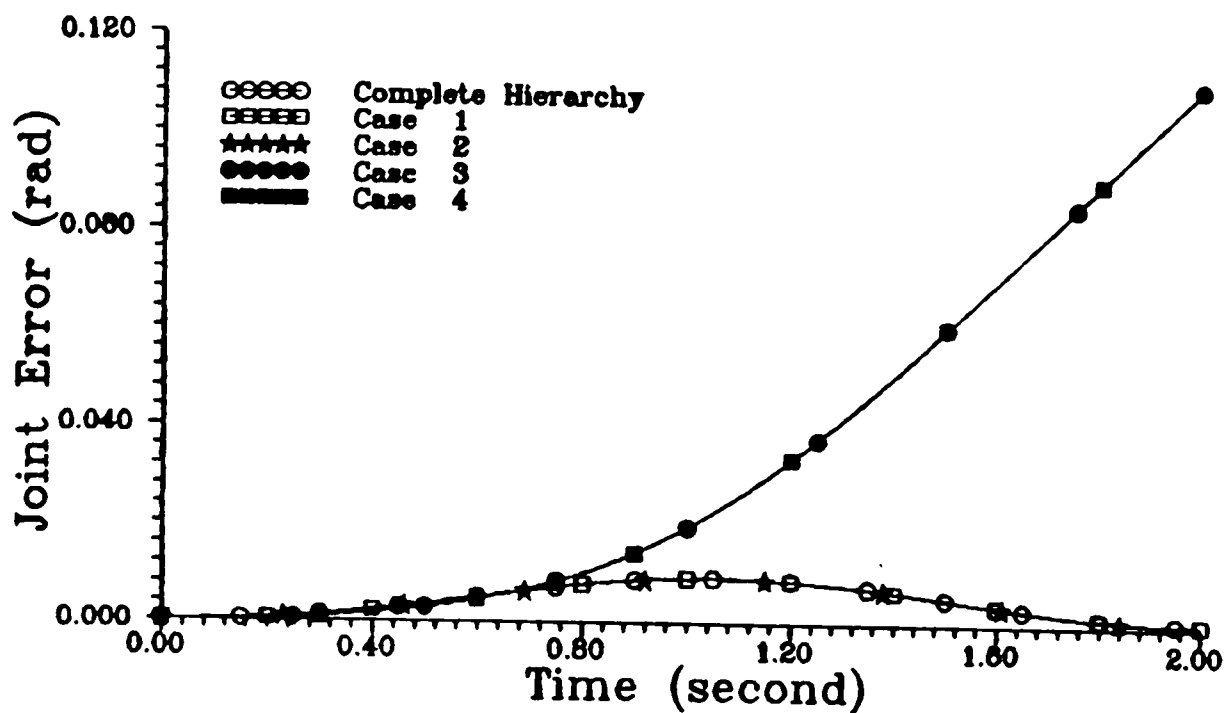


FIGURE 6.27 Joint 3 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 1

### Joint 1 Tracking Error

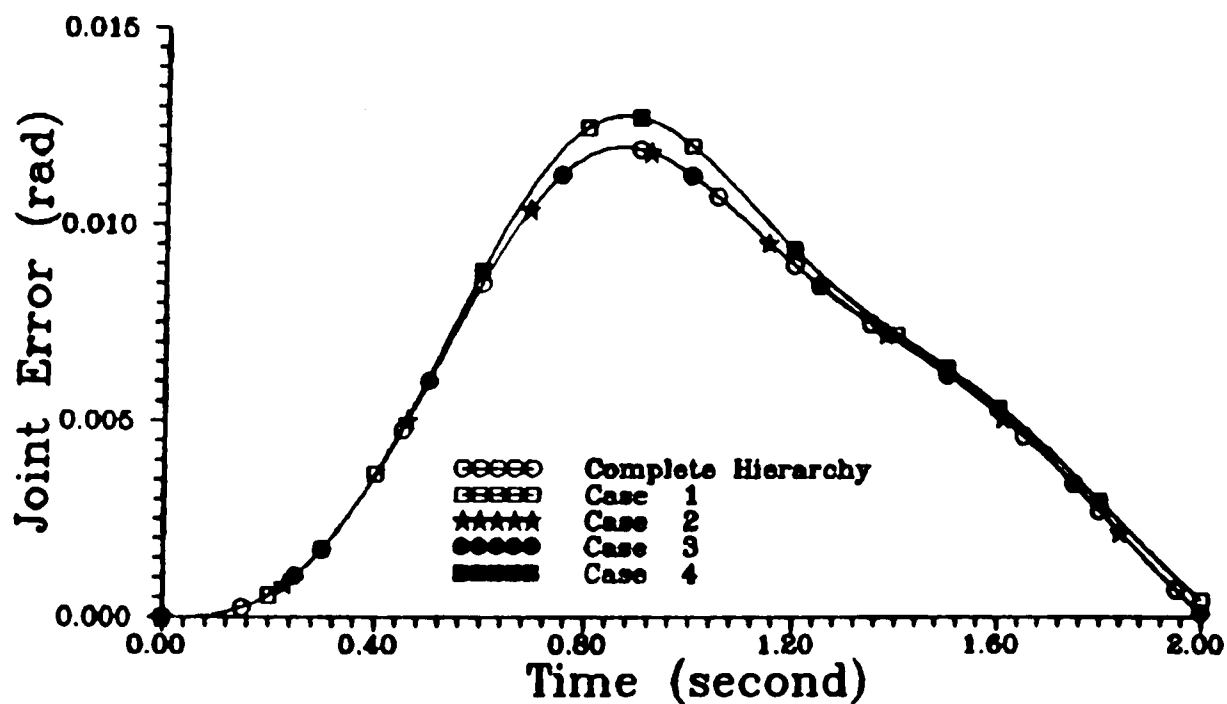


FIGURE 6.28 Joint 1 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 2



### Joint 2 Tracking Error

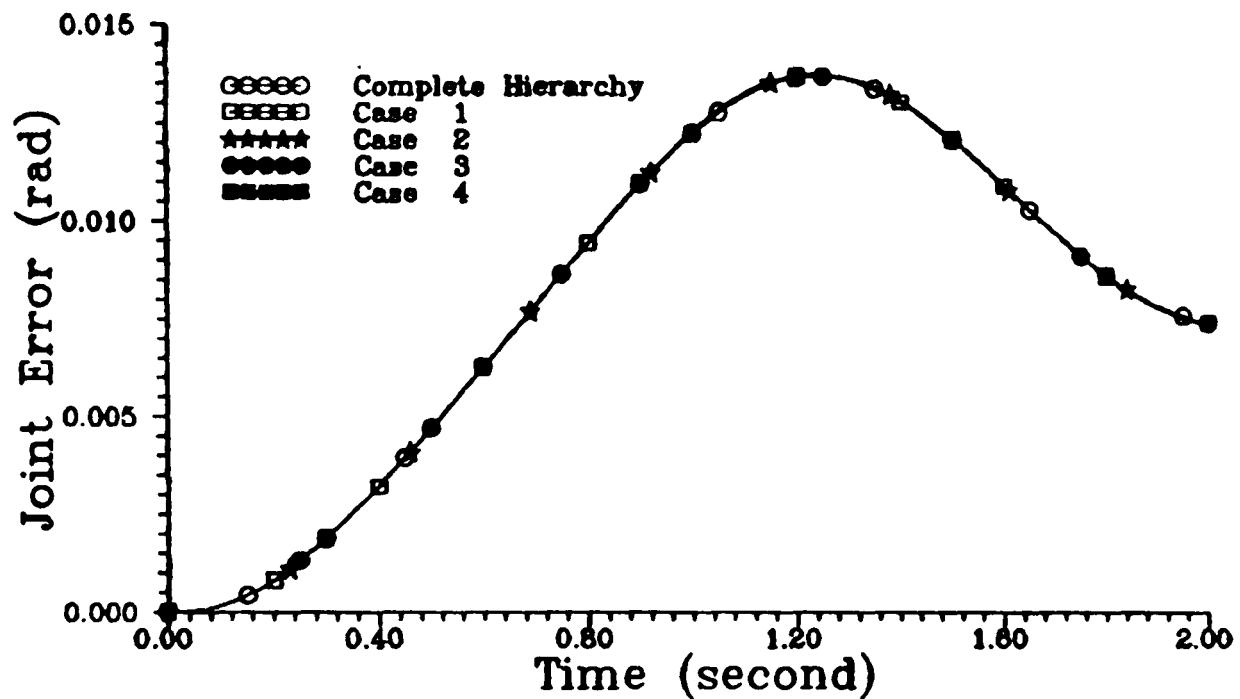


FIGURE 6.29 Joint 2 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 2

### Joint 3 Tracking Error

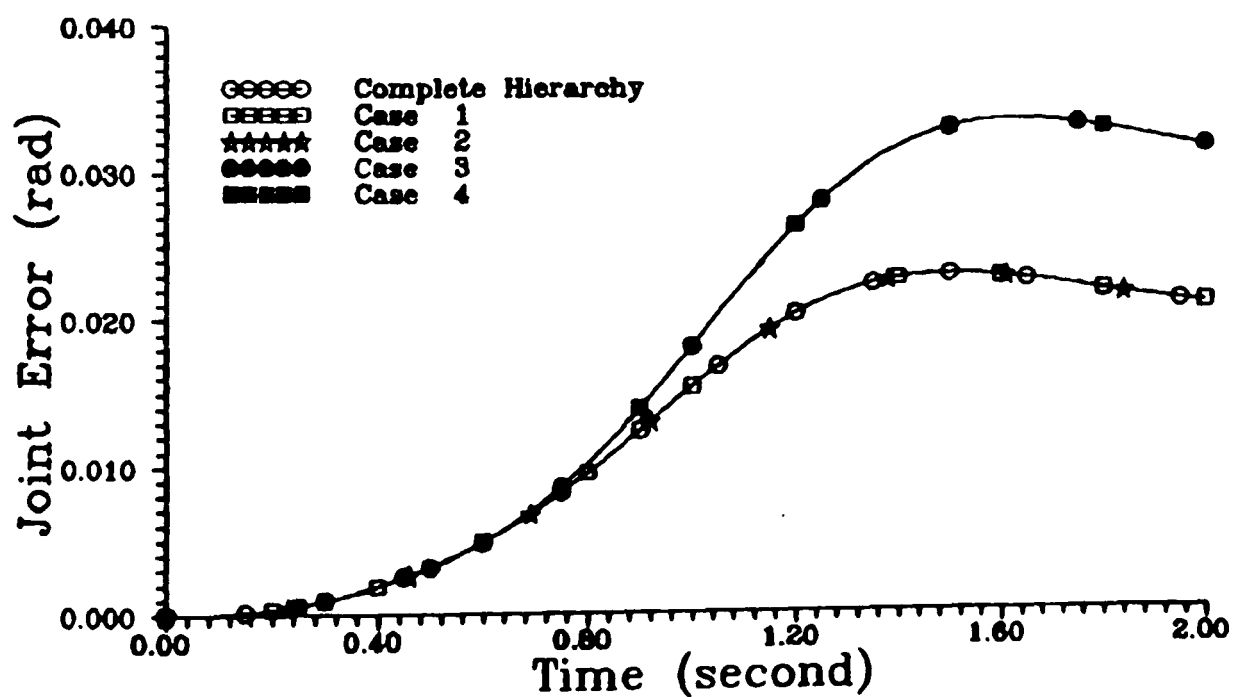


FIGURE 6.30 Joint 3 Tracking Error Under Different Structural Perturbation For Hierarchical Control – Method 2

The simulation results show that when there is an interruption between the coordinator and the subsystem, the tracking error for the uncoordinated subsystem is larger for the robot manipulator using the first method compared to the same robot manipulator using the second control method. This is because, for the manipulator using the first method, the uncoordinated uncertain subsystems are being controlled only by the local linear controller which is incapable of compensating the effect of the uncertainties and nonlinearities present in the subsystems and, in some cases, it may cause instability if the linear controller gain is not large enough (see Figure 6.27). Whereas for the manipulator using the second method, the local controller for the uncoordinated subsystem behaves like the decentralized local control law as presented in Chapter 4. Thus, for the second method, the nonlinear local controller is still capable of compensating the effect of the uncertainties and nonlinearities present in the corresponding uncoordinated subsystems. Hence, the response of the robot manipulator using the second method is better than that obtained using the first method under the structural perturbation.

#### 6.5.5 Effect Of Varying The Sampling Interval At The Upper level

It is a great advantage if the computation effort at the upper level of the control hierarchy (computation of the coordination functions) can be reduced. In the following simulations, the effect of reducing the computation frequency of the coordination functions at the upper level on the tracking performance of the robot manipulator using the two hierarchical control strategies is studied.

In these simulations, the sampling interval at the first level has been defined at 0.01s. At the second level, the following cases have been considered :

Case 1 : The sampling interval for level two is at 0.03s

Case 2 : The sampling interval for level two is at 0.05s

Case 3 : The sampling interval for level two is at 0.10s

For the robot manipulator using the first method, the resulting tracking responses are as shown in Figures 6.31 through 6.37. The simulations show that the control inputs to the joint actuators will become oscillatory and the tracking performance of the robot manipulator deteriorates as the sampling interval at the upper level is increased.

A much better performance for the robot manipulator can be achieved using the second hierarchical control method (Figures 6.38 through 6.48). The figures show that the tracking performance of the robot manipulator using the second method is not significantly affected by the increase in the sampling interval at the second level of the control hierarchy. The control inputs are smooth as compared to that of using the first hierarchical control method (figures 6.36 and 6.37).

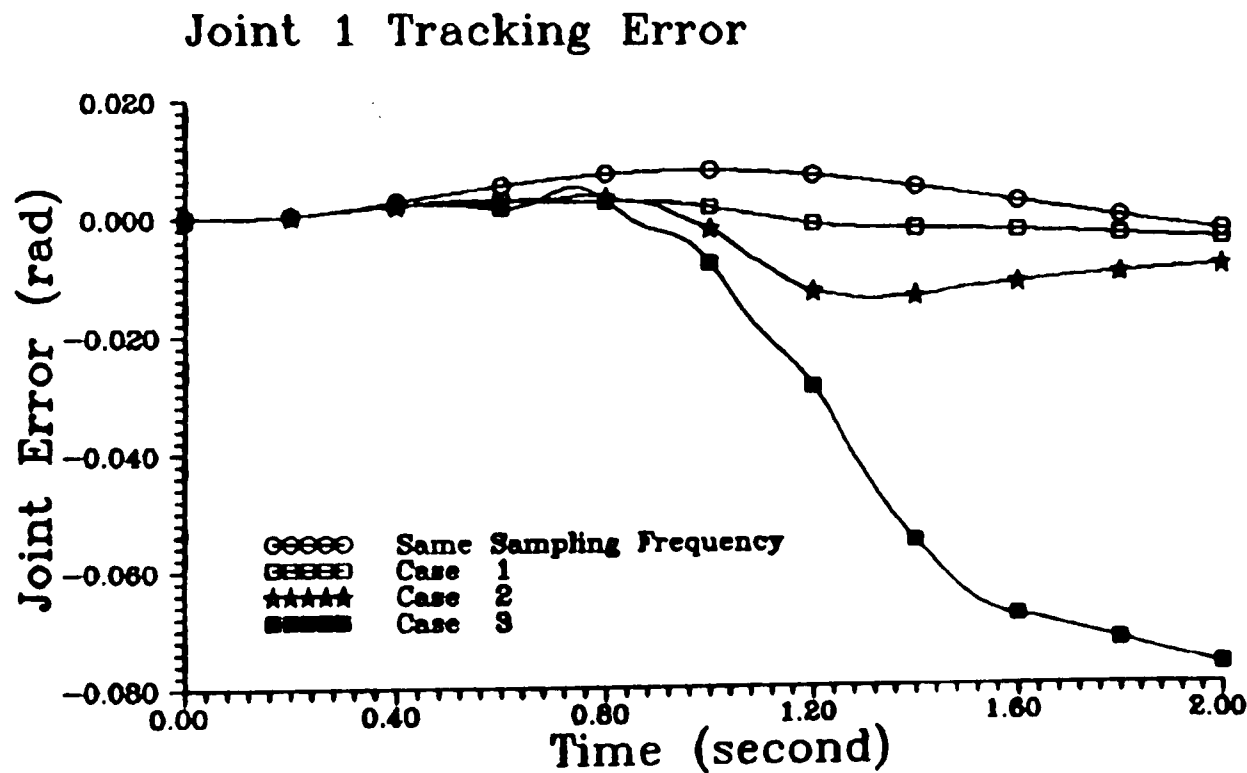


FIGURE 6.31 Joint 1 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 1

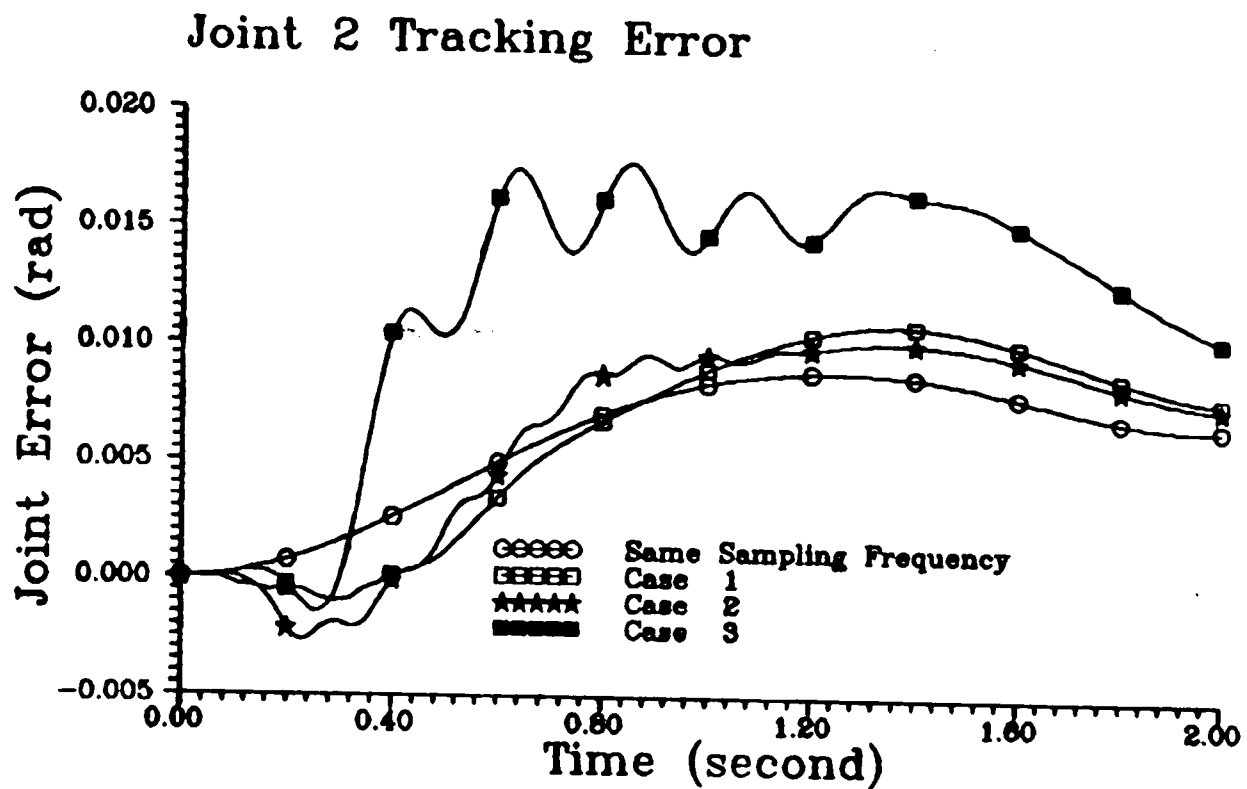


FIGURE 6.32 Joint 2 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 1

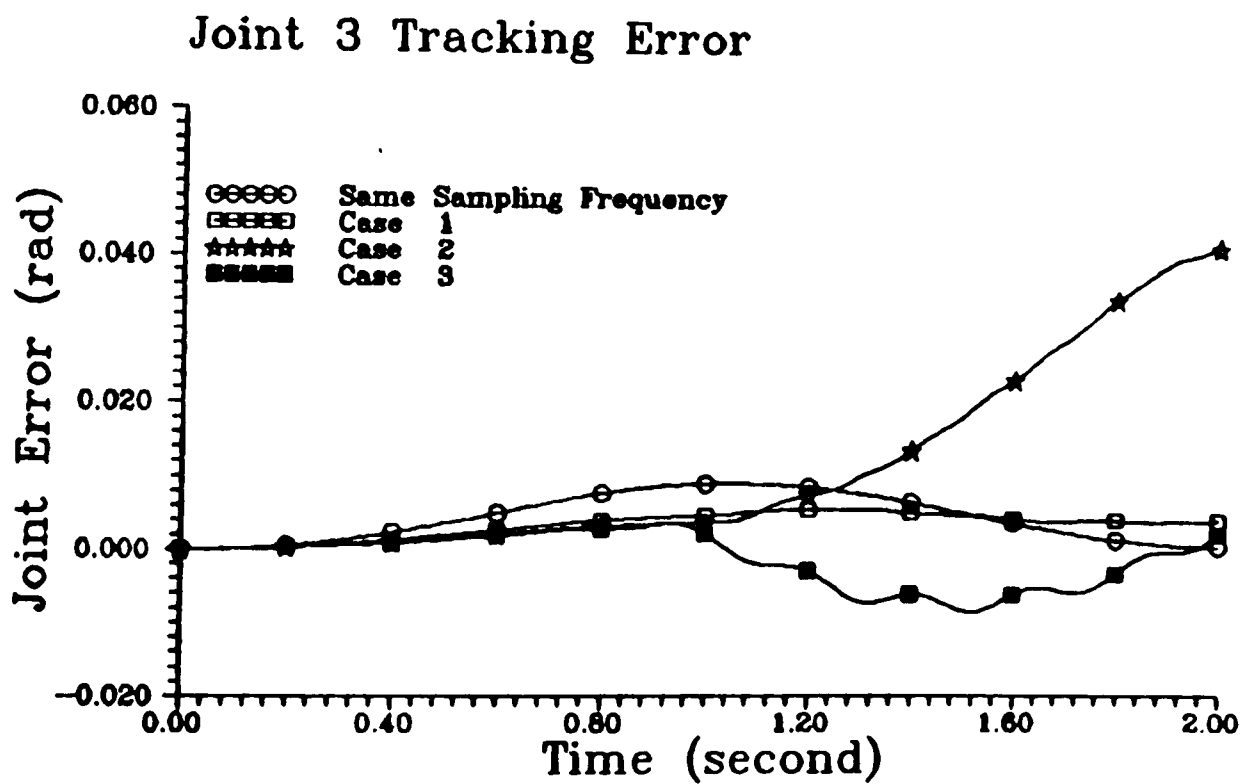


FIGURE 6.33 Joint 3 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 1

### Joint Tracking Response For 10 Kg Load

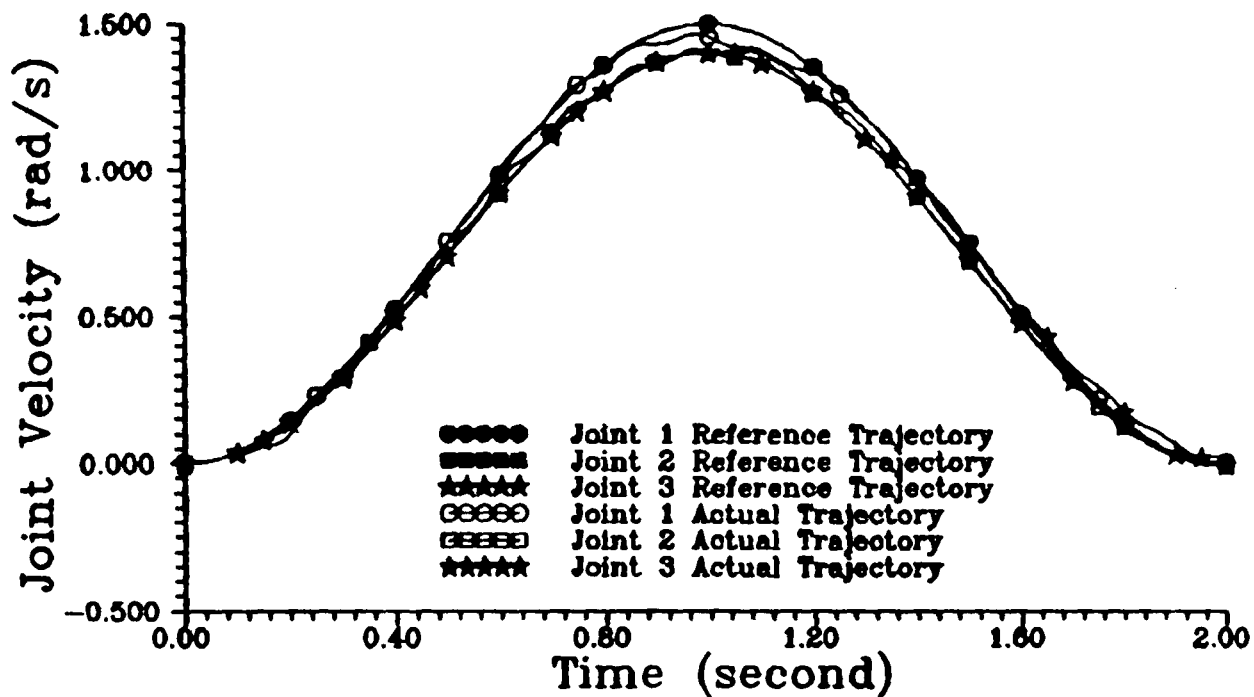


FIGURE 6.34 a : Velocity Tracking Response When The Sampling Rates Of The Coordinator Is At 0.05s – Method 1

### Joint Tracking Response For 10 Kg Load

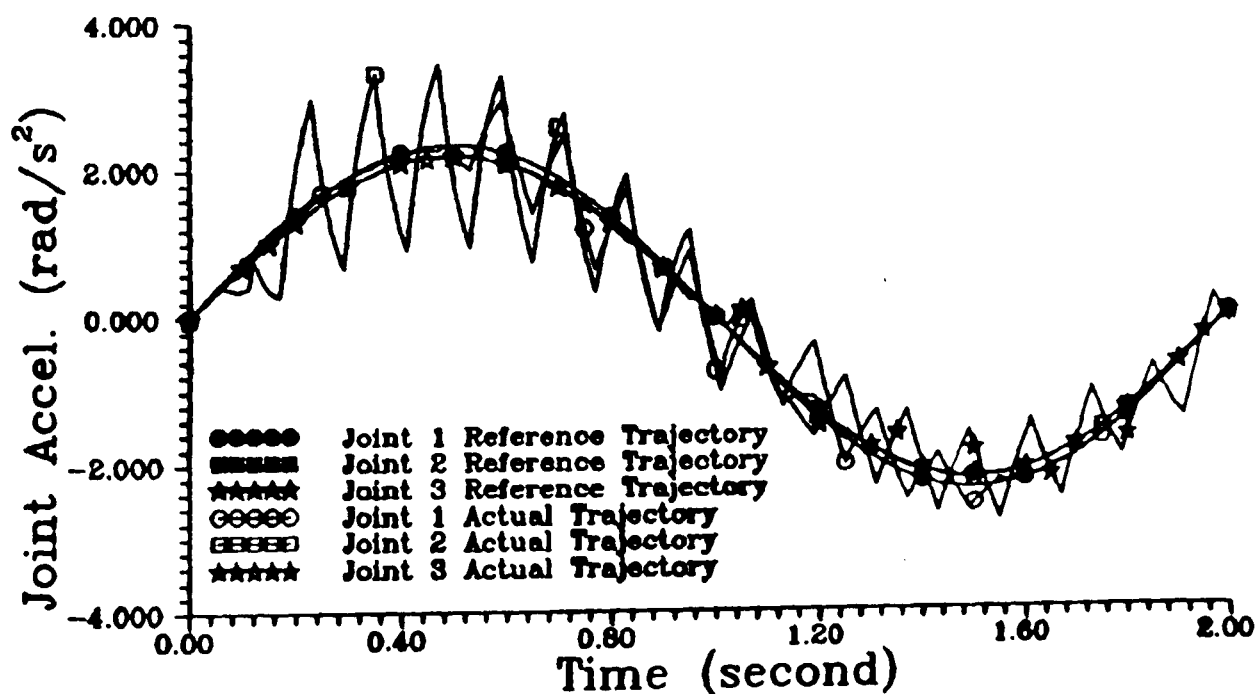


FIGURE 6.34 b : Acceleration Tracking Response When The Sampling Rates Of The Coordinator Is At 0.05s – Method 1

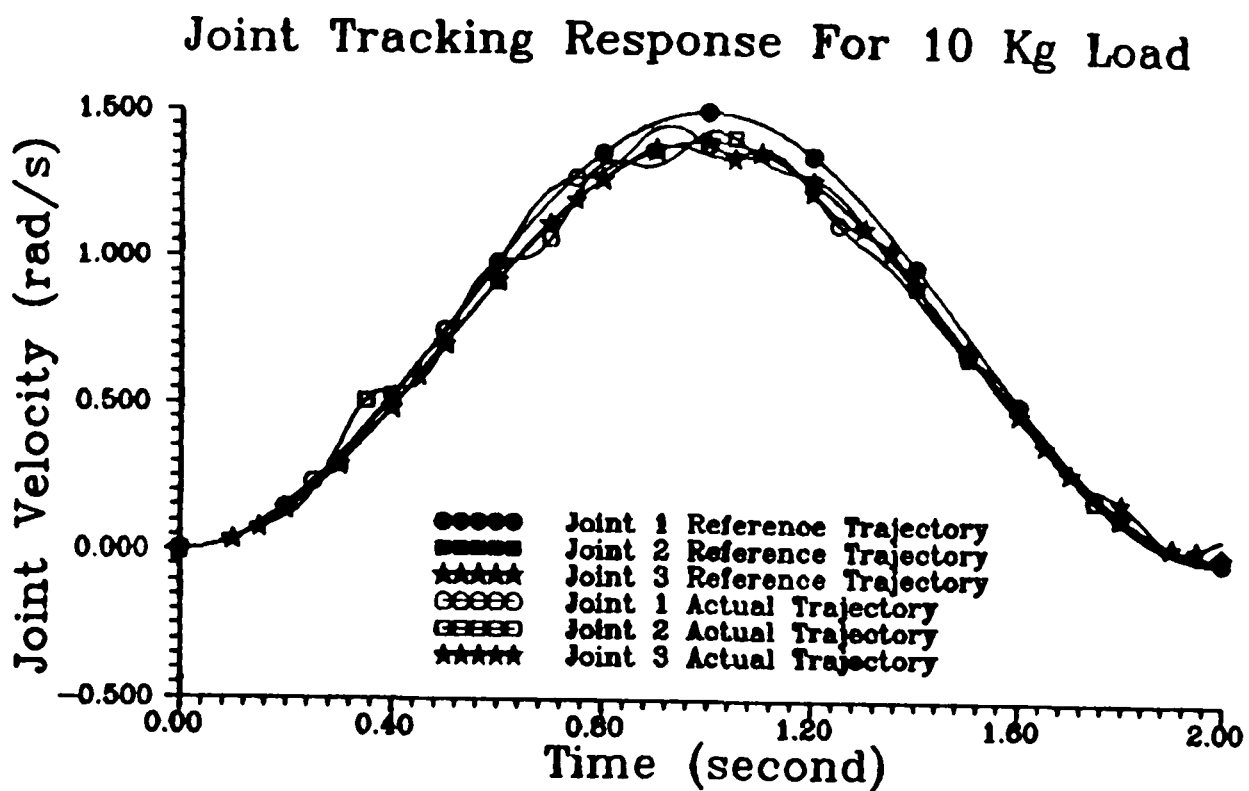


FIGURE 6.35 a : Velocity Tracking Response When The Sampling Rates Of The Coordinator Is At 0.1s – Method 1

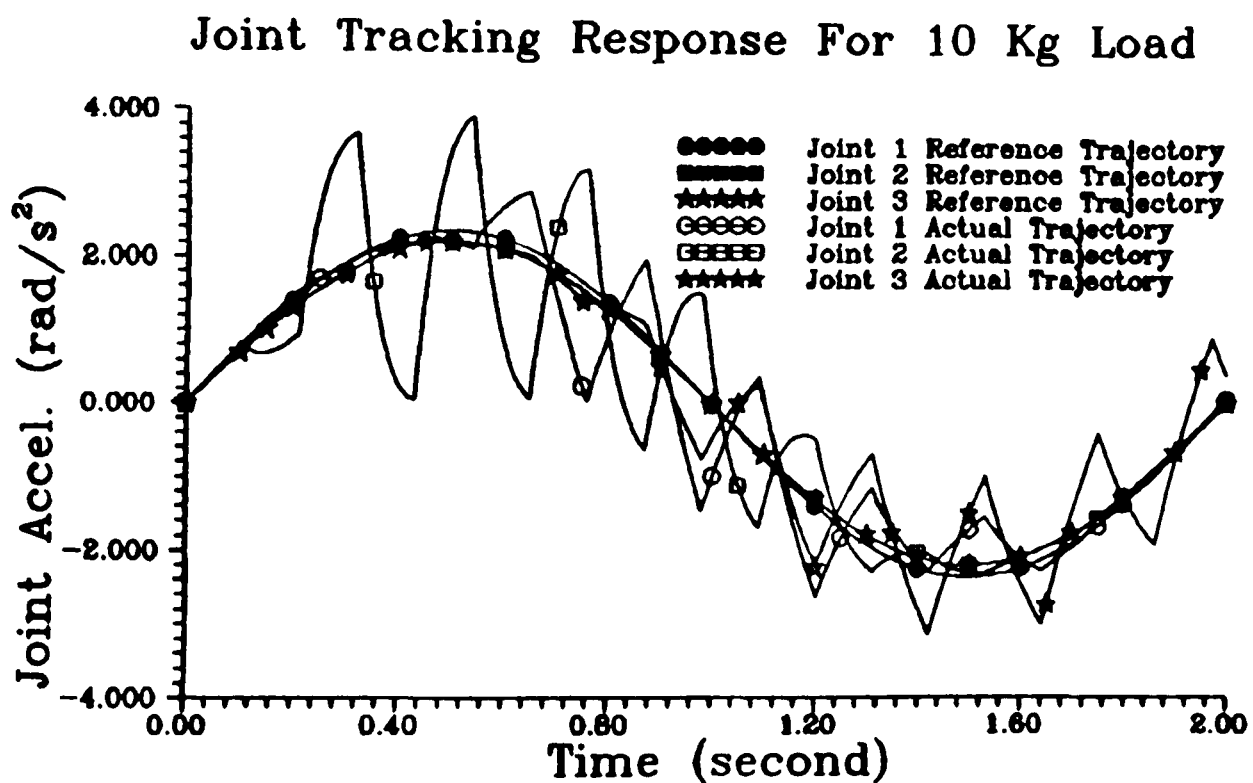


FIGURE 6.35 b : Acceleration Tracking Response When The Sampling Rates Of The Coordinator Is At 0.1s – Method 1

Control Input For 10 Kg Load

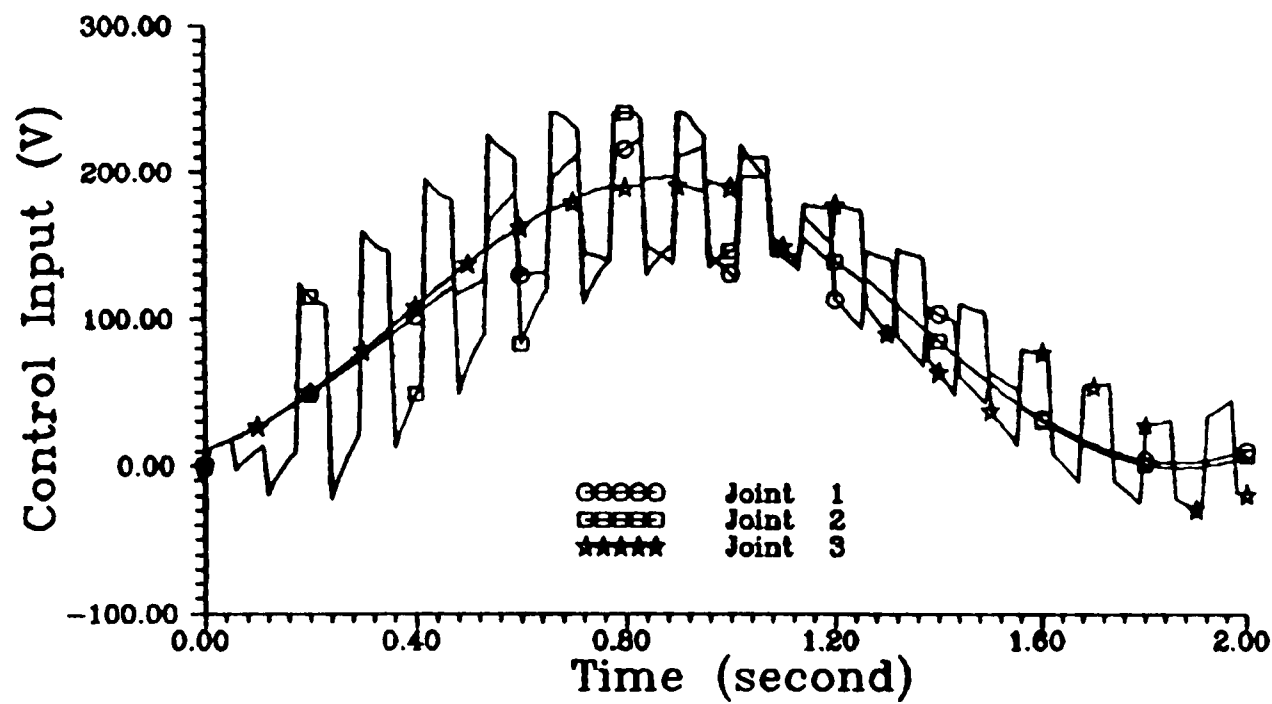


FIGURE 6.36 : Joint Control Inputs When The Sampling Rate Of The Coordinator Is At 0.05s – Method 1

Control Input For 10 Kg Load

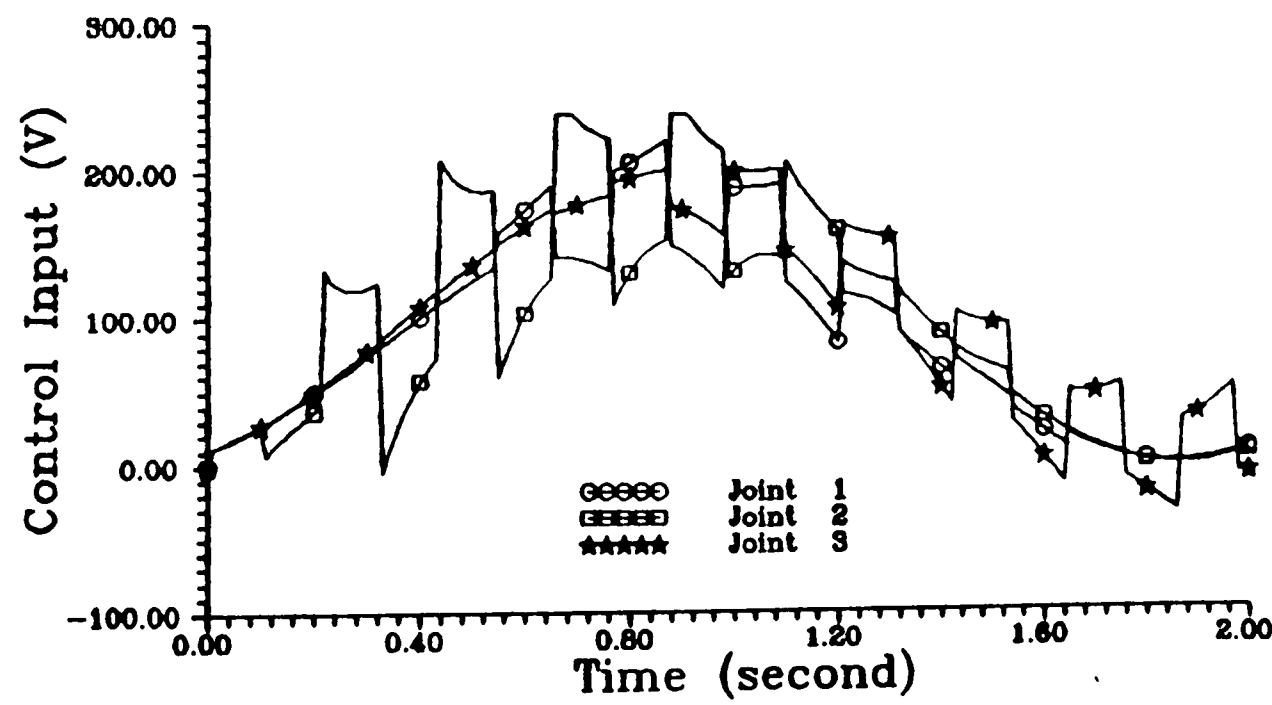


FIGURE 6.37 : Joint Control Inputs When The Sampling Rate Of The Coordinator Is At 0.1s – Method 1

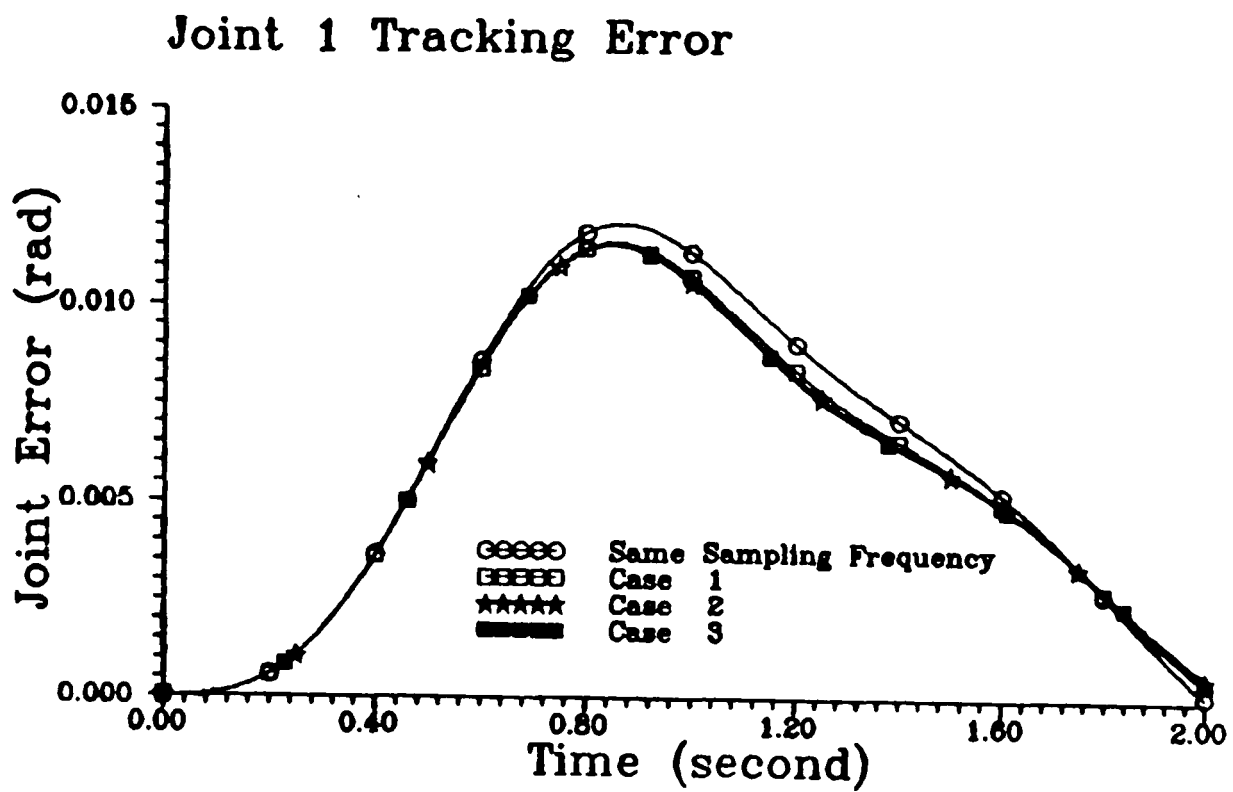


FIGURE 6.38 : Joint 1 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 2

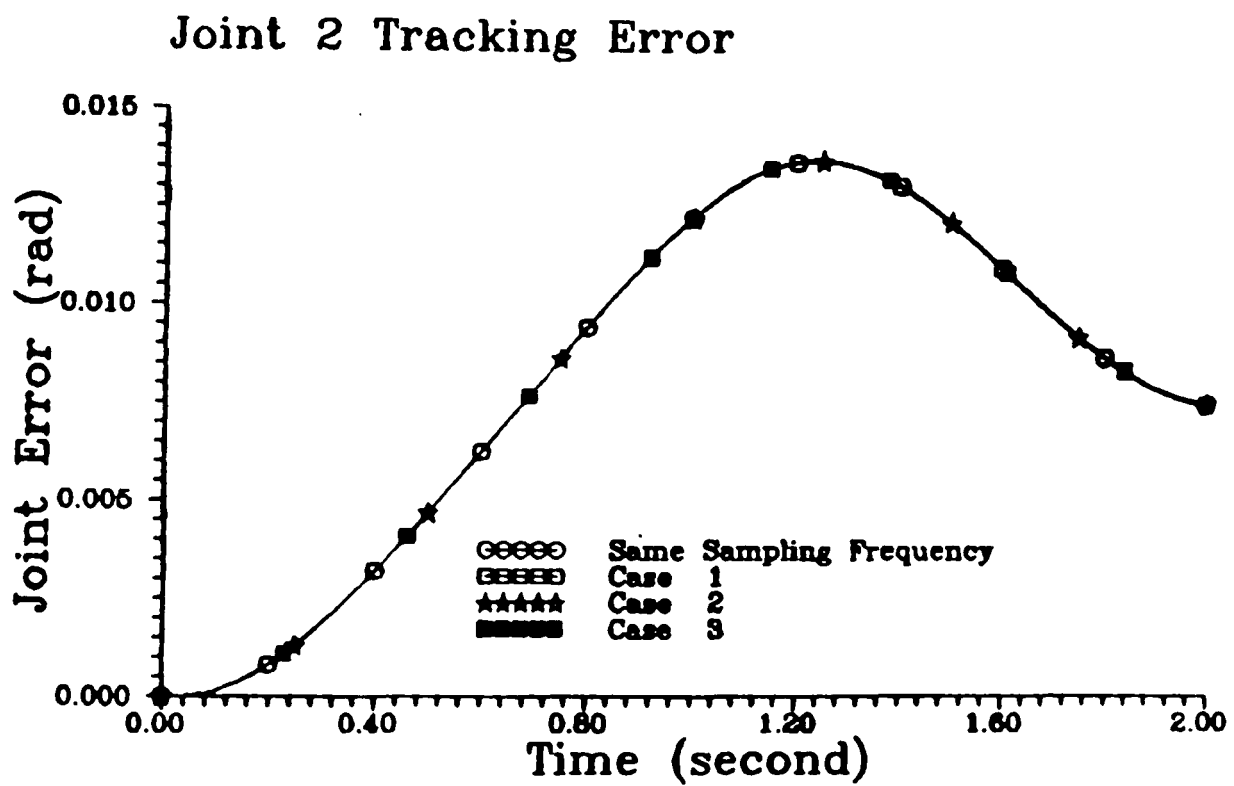


FIGURE 6.39 : Joint 2 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 2



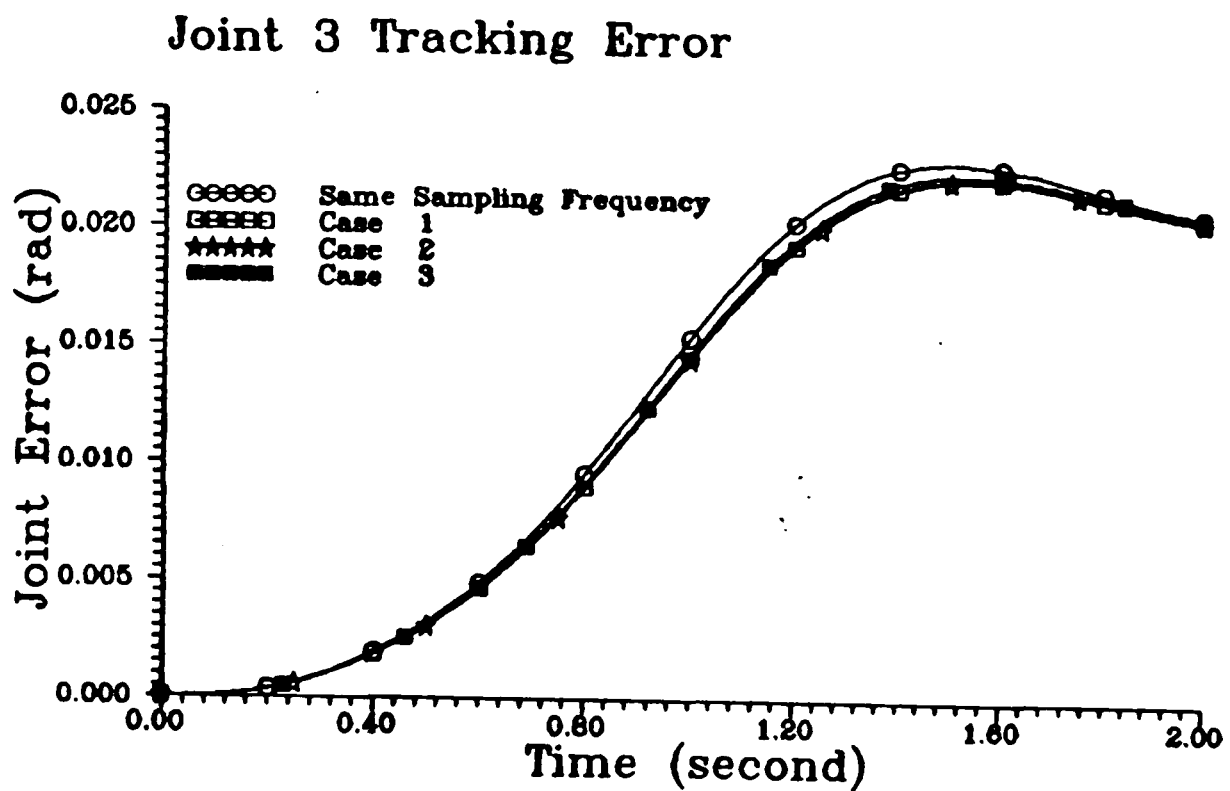


FIGURE 6.40 : Joint 3 Tracking Error Under Different Rates Of Computation Of The Coordinator – Method 2

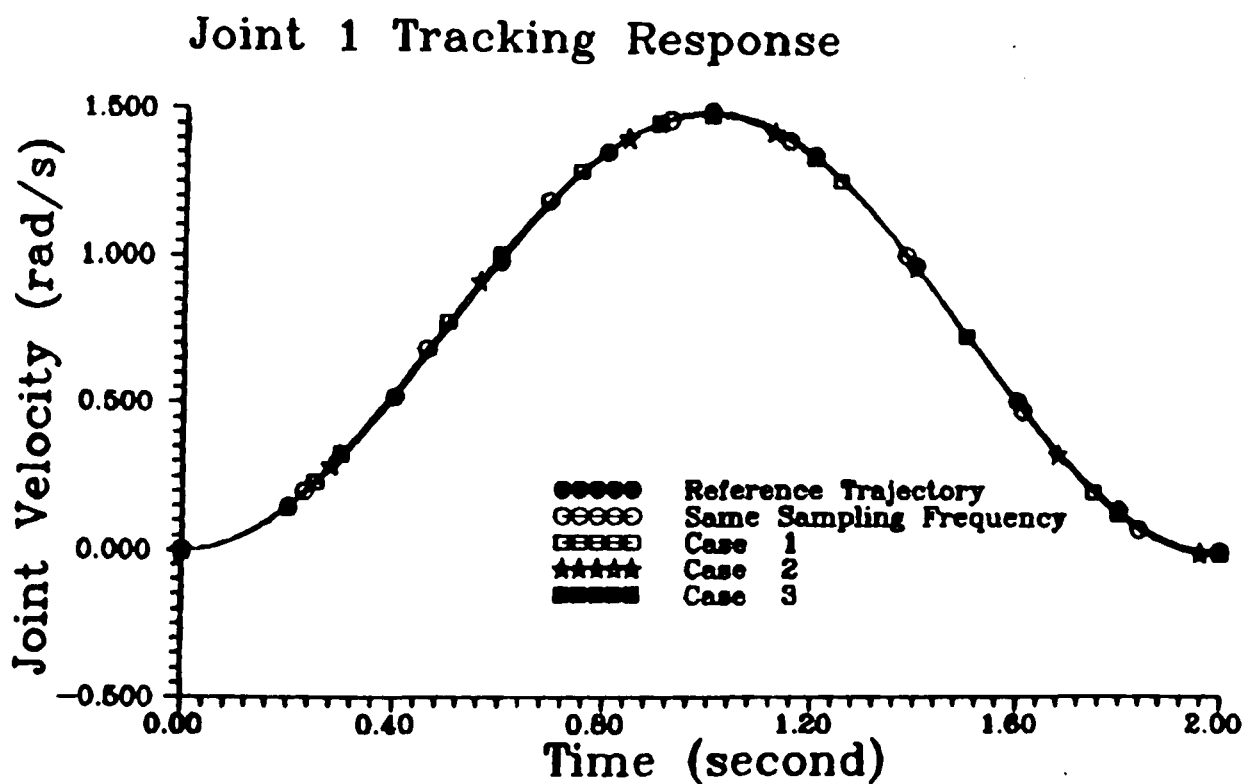


FIGURE 6.41 : Joint 1 Velocity Response Under Different Rates Of Computation Of The Coordinator – Method 2

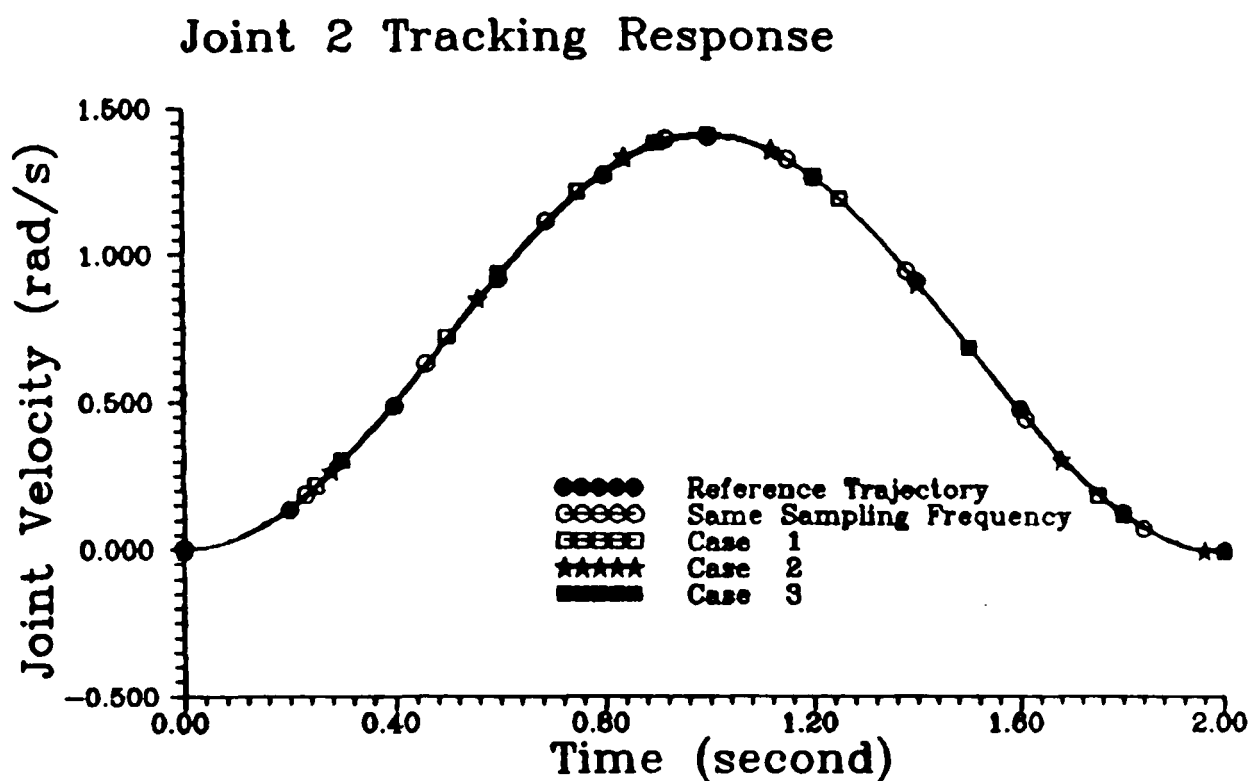


FIGURE 6.42 : Joint 2 Velocity Response Under Different Rates Of Computation Of The Coordinator – Method 2

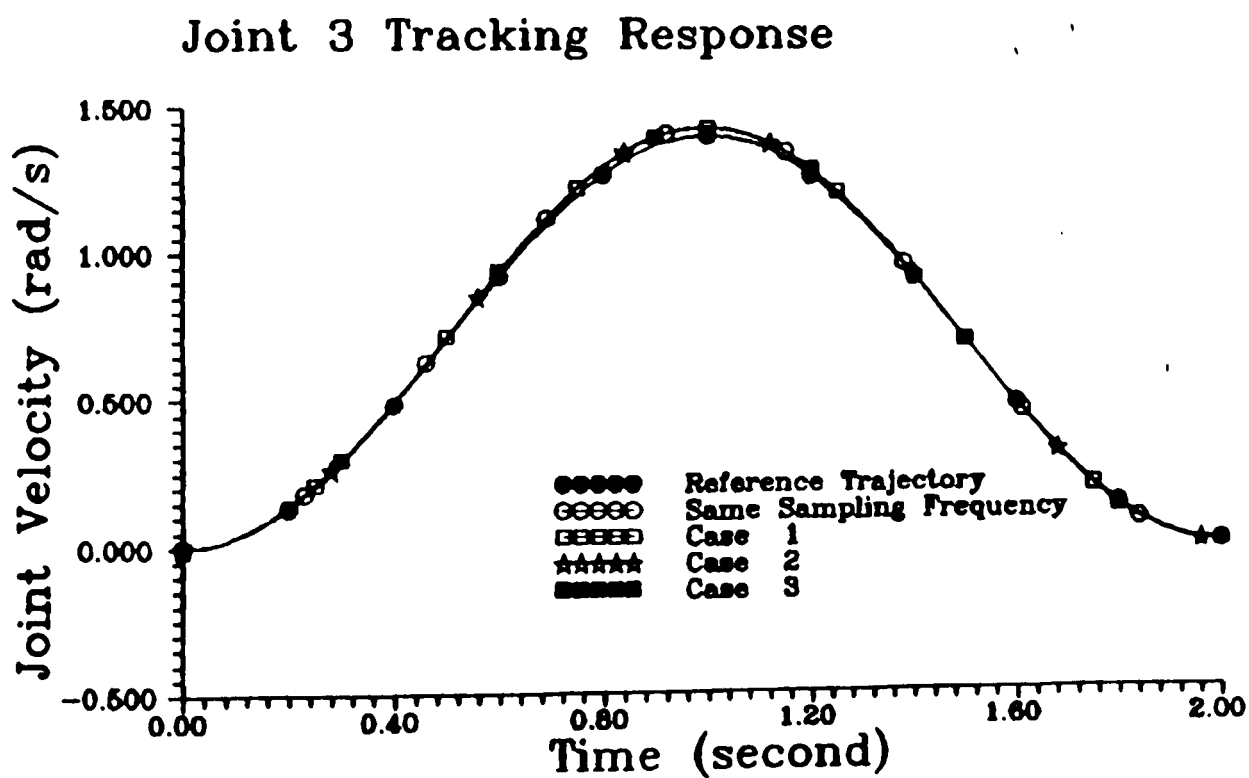


FIGURE 6.43 : Joint 3 Velocity Response Under Different Rates Of Computation Of The Coordinator – Method 2

### Joint 1 Tracking Response

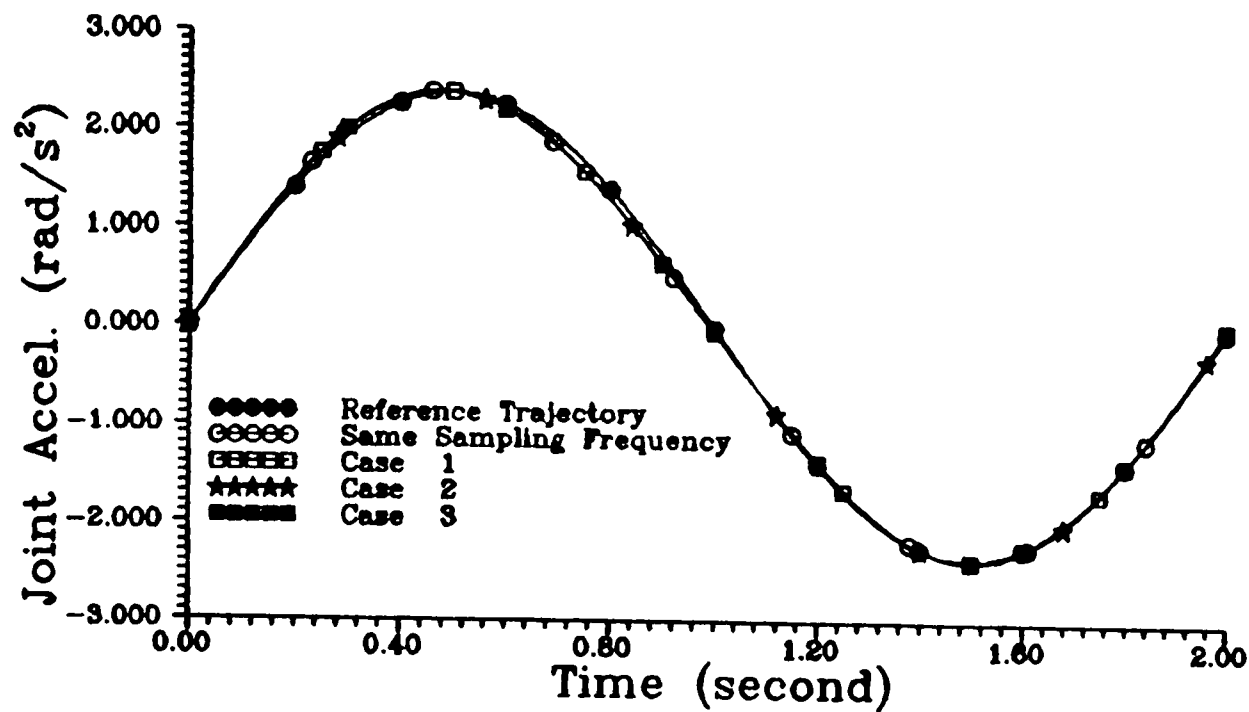


FIGURE 6.44 : Joint 1 Acceleration Response Under Different Rates Of Computation Of The Coordinator – Method 2

### Joint 2 Tracking Response

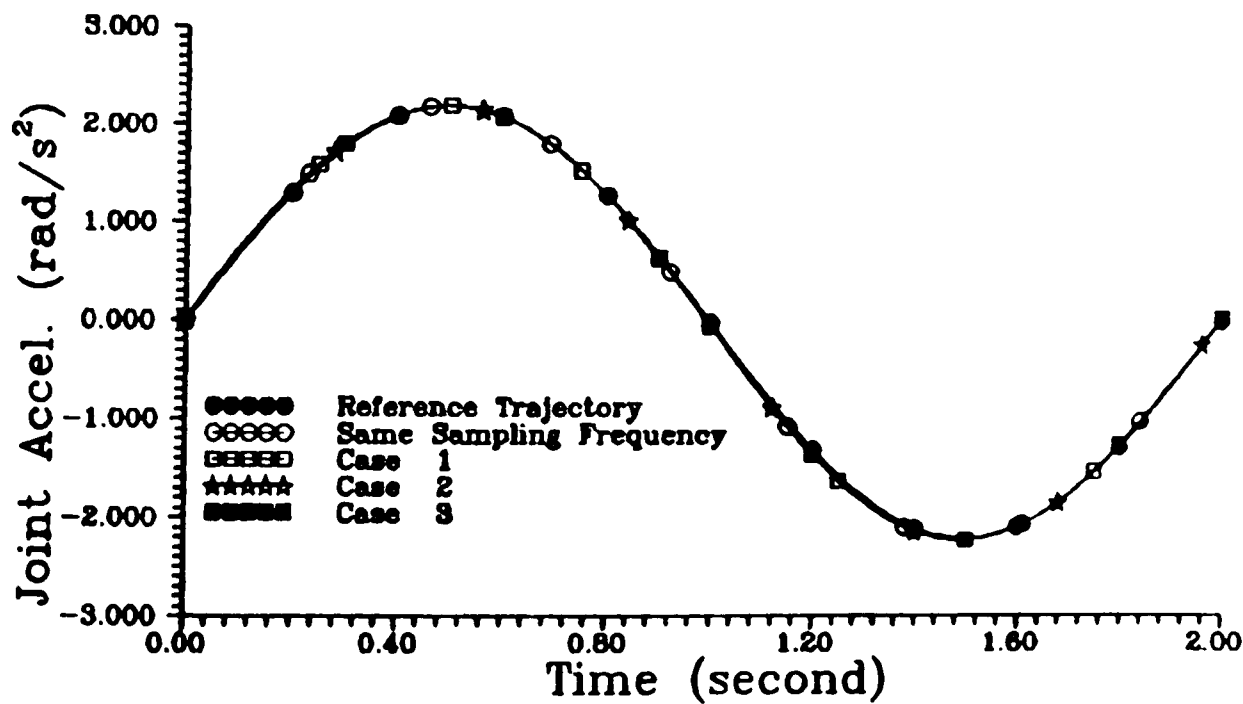


FIGURE 6.45 : Joint 2 Acceleration Response Under Different Rates Of Computation Of The Coordinator – Method 2

### Joint 3 Tracking Response

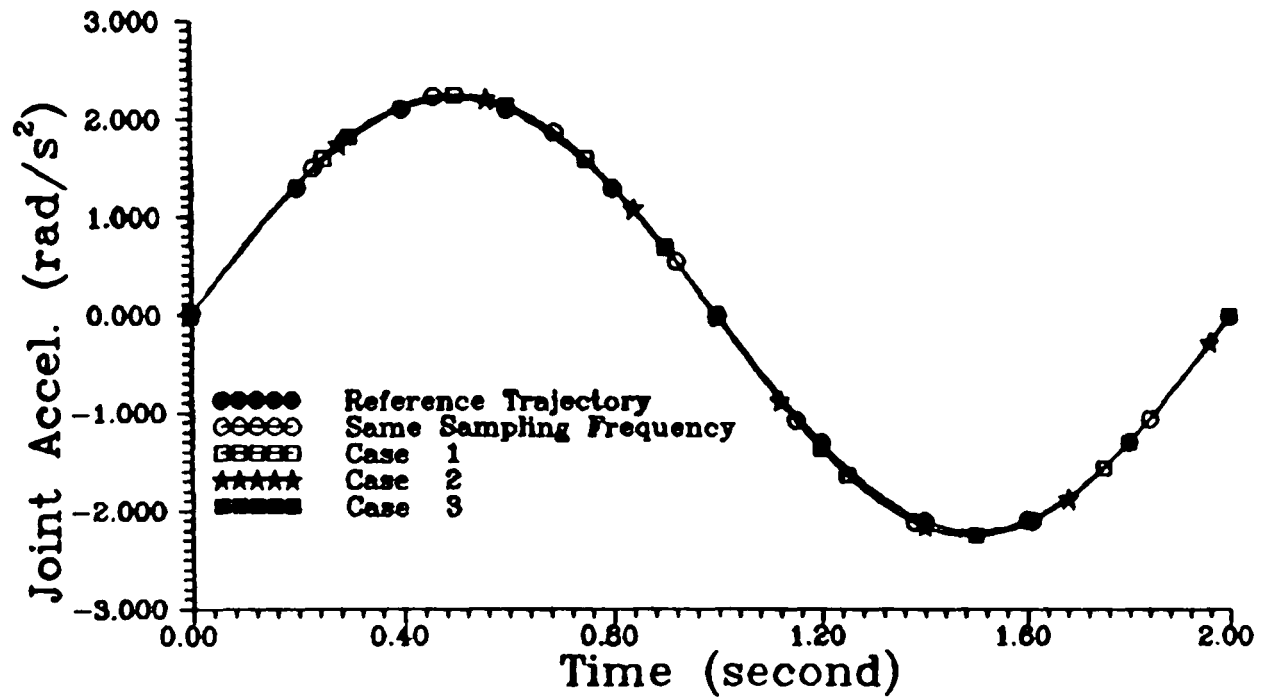


FIGURE 6.46 : Joint 3 Acceleration Response Under Different Rates Of Computation Of The Coordinator – Method 2

### Control Input For 10 Kg Load

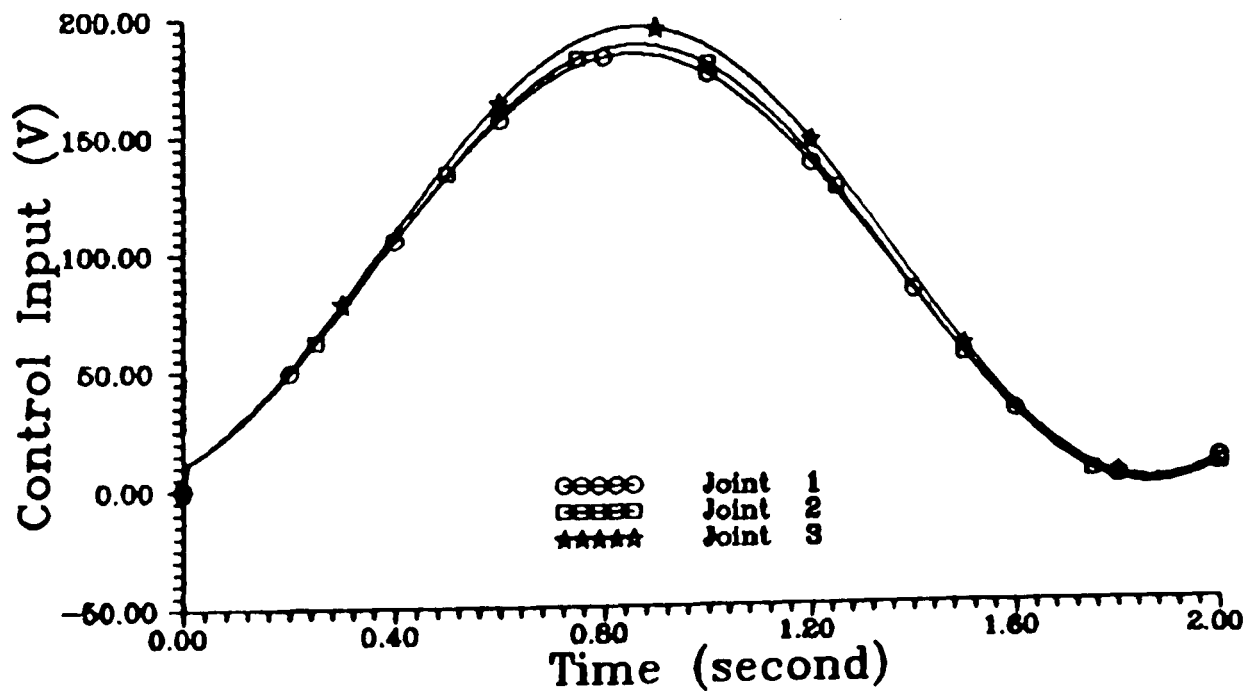


FIGURE 6.47 : Joint Control Inputs When The Sampling Rate Of The Coordinator Is At 0.05s – Method 2

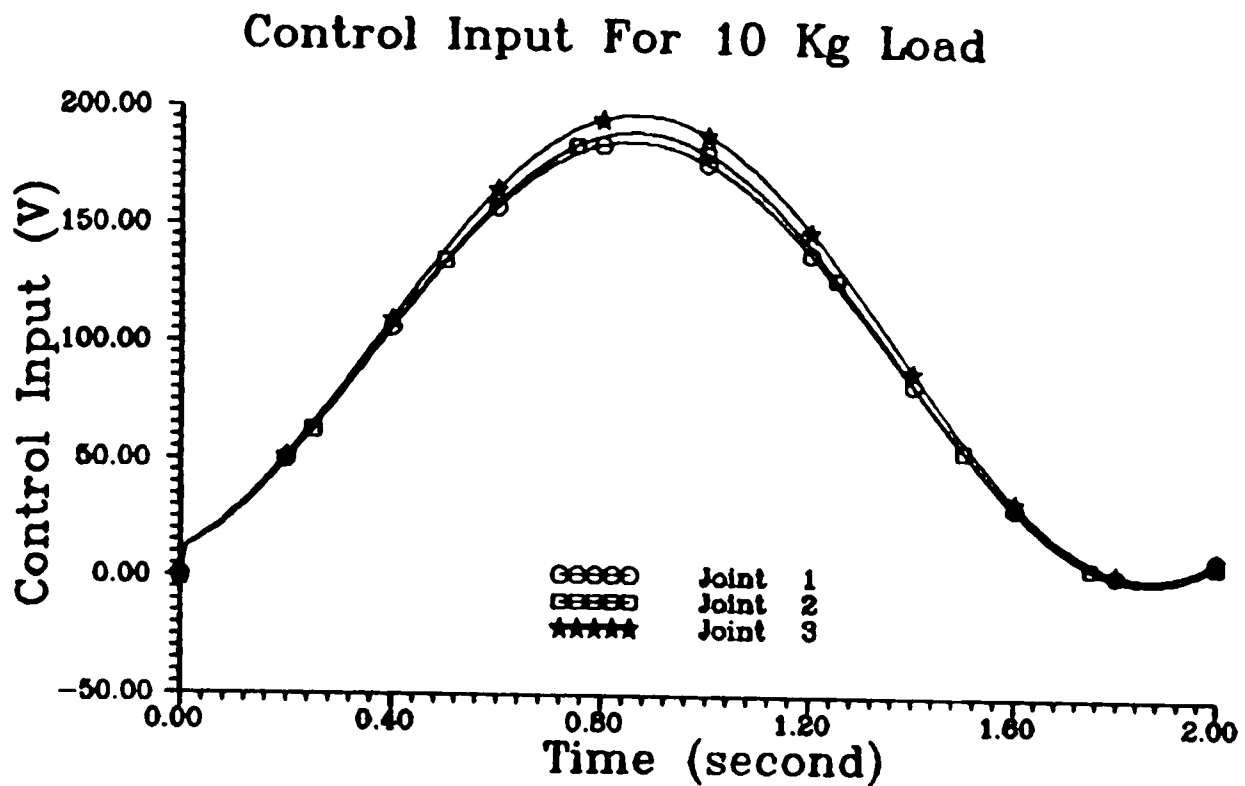


FIGURE 6.48 : Joint Control Inputs When The Sampling Rate  
Of The Coordinator Is At 0.1s – Method 2

The simulations show that in order for the first method to have a good tracking accuracy, the coordinator must be computed and send the compensating signal to each subsystem at the same sampling rate as the lower level. This is because each subsystem at the lower level cannot compensate or reduce the uncertainties and nonlinearities present in the subsystem through its own local controller, and the compensating signals must be available at every sampling interval of the subsystem. Whereas for the second method, the local controller at the lower level can compensate the effect of the uncertainties and nonlinearities present in the corresponding subsystem, and the signal from the coordinator is only needed to correct the effect of the interconnection functions between the subsystems. From the simulation results it is shown that, for the second method, the correcting signals at the upper level may be computed at a sampling interval slower than that of the lower level. Thus, the computation of the coordinator can be reduced. This will result in the reduction of the overall on-line computation of the controller as compared to the first method. Hence, in this case, the second

method is better than the former technique.

### 6.5.6 Comparison Between The Proposed Control Methods

In this final set of simulations, the performance of the proposed hierarchical control method, the decentralized global control method and the centralized control approach as used in the previous chapter are compared. Here, only the second hierarchical control method is employed. This is because the first hierarchical control strategy is similar to the decentralized global control method in terms of their performance.

For the decentralized and hierarchical control laws, the following feedback gains have been used :

$$\begin{aligned}
 \text{Subsystem 1} : \quad K_1 &= \begin{bmatrix} -3.5781 & 111.004 & -0.065 \end{bmatrix} \\
 \text{Subsystem 2} : \quad K_2 &= \begin{bmatrix} -400.0979 & -58.9978 & -14.4766 \end{bmatrix} \\
 \text{Subsystem 3} : \quad K_3 &= \begin{bmatrix} -2.5201 & 128.5255 & 0.2785 \end{bmatrix}
 \end{aligned} \tag{6.82}$$

which correspond to the following closed-loop poles :

$$\begin{aligned}
 \text{Subsystem 1} : \quad & -0.4 \quad , \quad -0.4 \quad , \quad -5.0 \\
 \text{Subsystem 2} : \quad & -5.0 \quad , \quad -5.0 \quad , \quad -10.0 \\
 \text{Subsystem 3} : \quad & -0.3 \quad , \quad -0.3 \quad , \quad -5.0 .
 \end{aligned}$$

In solving the matrix Lyapunov equation (equation 6.24), the positive definite symmetrical matrix  $Q_i$  was taken to be a 3x3 identity matrix for each subsystem. The following values of  $\epsilon_i$  were used :

$$\epsilon_1 = 0.5 \ ; \ \epsilon_2 = 0.3 \ ; \ \epsilon_3 = 0.5 \ .$$

For the centralized control method, the same controller used in the subsection 5.4.5 was utilized. The following feedback gain was used for the centralized control method :

$$K = \text{diag} [ K_1 , K_2 , K_3 ] \ .$$

The matrix  $Q$  has been chosen as a 9x9 identity matrix for the centralized controller design, and  $\epsilon = 1.0$  has been used.

In the simulations, the robot manipulator was required to move from an initial position of  $\theta(0) = [ -0.8 , -1.5 , -0.5 ]^T$  radians to the final position of  $\theta(\tau) = [ 1.0 , 0.2 , 1.2 ]^T$  radians in  $\tau = 2$  seconds, while carrying a 10 Kg. load.

For comparison purposes, the responses for each case are plotted on the same graph. Figures 6.49, 6.50, and 6.51 illustrate the tracking performance of the three dof robot manipulator under the various control methods considered for joint 1 , joint 2, and joint 3, respectively.

The figures illustrate that the performance of the decentralized global control law is a little better than that of the second hierarchical control technique, but comparable to the centralized control method. It should be noted that the simulation for each controller is not under the same conditions. In terms of computation time, the centralized control method is the worst among the three approaches considered. The main advantage of the hierarchical control method over the decentralized global control technique is that the computation time needed to compute the controller can be reduced by increasing the sampling time at the upper level of the control hierarchy.

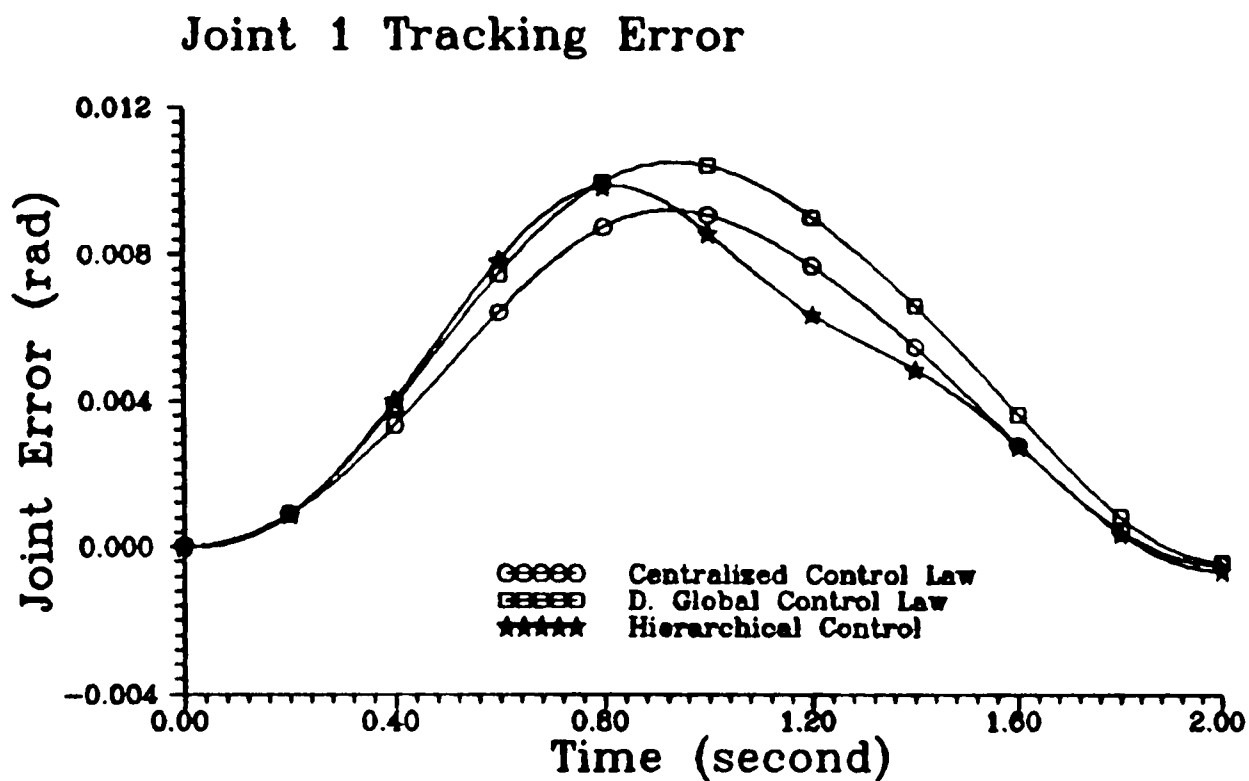


FIGURE 6.49 : Joint 1 Tracking Error Using Decentralized, Hierarchical, And Centralized Control Methods

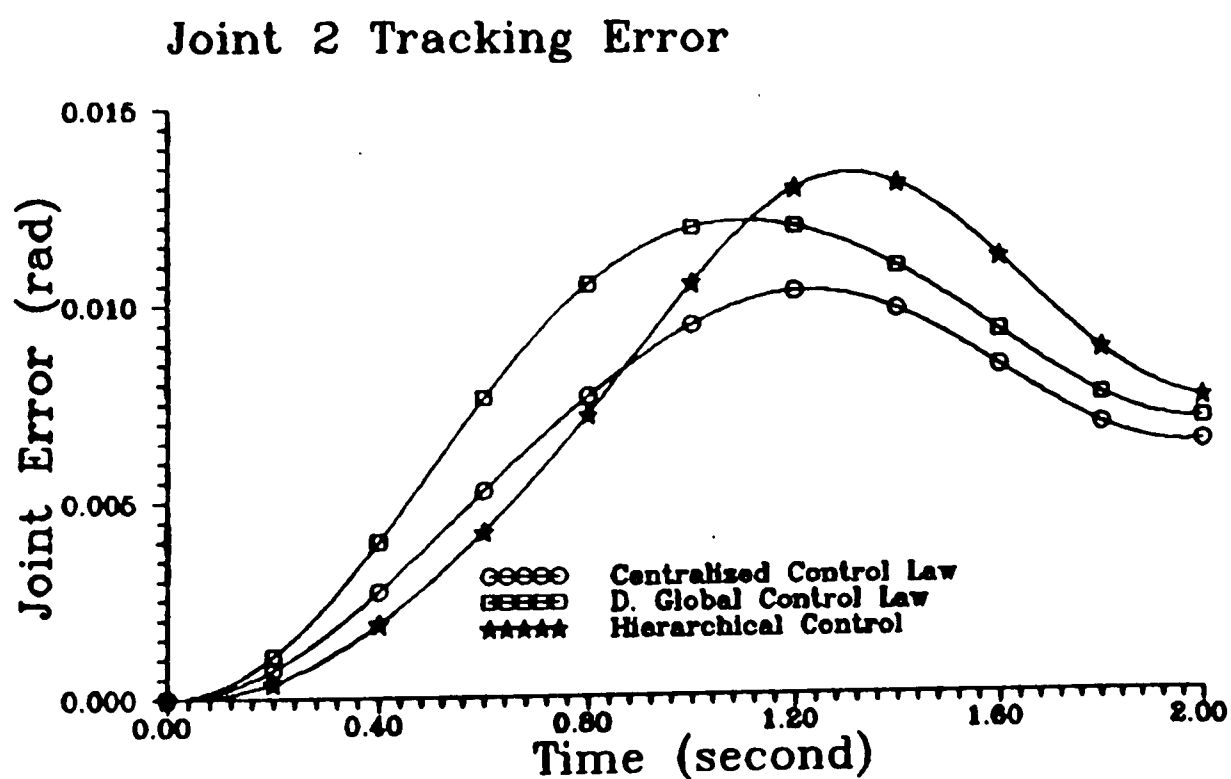


FIGURE 6.50 : Joint 2 Tracking Error Using Decentralized, Hierarchical, And Centralized Control Methods



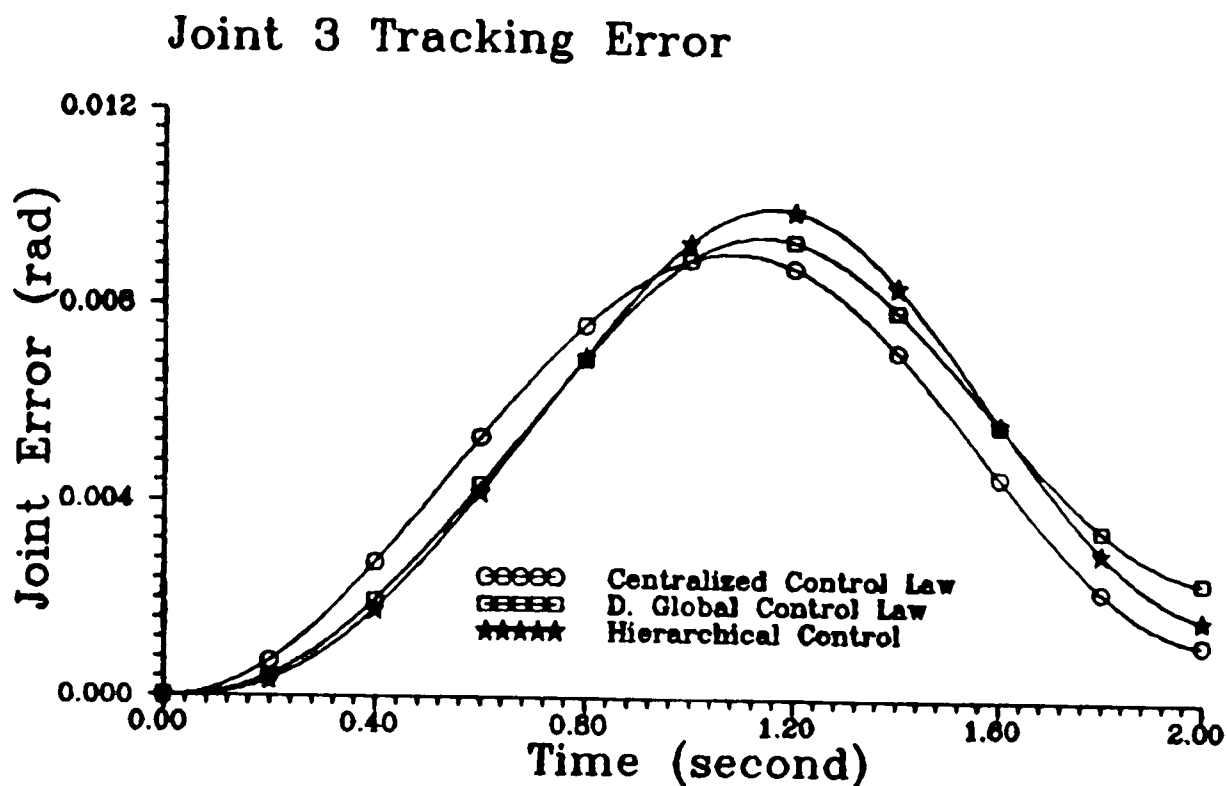


FIGURE 6.51 : Joint 3 Tracking Error Using Decentralized, Hierarchical, And Centralized Control Methods

## 6.6 CONCLUSION

Two two-level hierarchical control approaches have been proposed for on-line tracking control of robot manipulators. Both methods are based on a deterministic approach, and on the assumption that the system uncertainties, the nonlinearities and the interactions between the subsystems are bounded and these bounds are known. For both methods, sufficient stability conditions for the overall system have been derived based on the stability of the subsystems. The overall system will become unstable if one of the subsystems is unstable. Thus, the stability of the overall system is dependent on the stability of each subsystem. The methods are simple and robust with regard to the uncertainties and the nonlinearities present in the system. It was shown that the numerical computation of the coordination function at the higher level for the second method is minimal compared to those methods mentioned at the beginning of this chapter, and thus requires less computational time. Hence, the proposed method is suitable for on-line control of large scale systems with fast dynamics such as robot manipulators.

## CHAPTER 7

### CONCLUSION AND SUGGESTIONS

#### 7.1 CONCLUSION

The research described is concerned with both modelling and control aspects of robot manipulators. Emphasis has been given to the formulation of decentralized and hierarchical control of robot manipulators based on a deterministic approach.

A framework for the formulation of a complete mathematical dynamic model of a DC motor actuated revolute robot manipulator in state variable form has been presented. The derived model of the robot manipulator comprises the dynamics of the mechanical linkage as well as the dynamics of the actuators. The formulation results in nonlinear time varying state equations which represent a more realistic model of the robot manipulator than a model with the drive torques/forces modelled as ideal pure torque/force sources, or as first order lags.

It is shown that the selection of an appropriate set of state variables for deriving the model is vital for synthesizing an advanced and robust controller in order to overcome the nonlinearities, uncertainties, and couplings present in the robot manipulator system. The condition at which the nonlinearities, uncertainties, and couplings may be compensated depends only on the input matrix which actually defines the points where the nonlinearities, the uncertainties, the couplings and the control enter into the system. The control input which enters the system through the input matrix can compensate directly the nonlinear and uncertain components, if the nonlinear, uncertain and coupling terms lie within the range space of the input matrix. If these terms lie outside the range space of the input matrix, then there is no control input that may compensate for them directly even if the uncertainties and the nonlinear

interconnection functions are known. For this reason, it is shown that the integrated dynamic model of the robot manipulator derived using the joint position, velocity and acceleration as the state variables is more suitable for synthesizing a robust controller for the robot manipulator than the mathematical model derived using the joint angle, velocity, and the armature current as the state variables.

The inclusion of the actuator dynamics into the robot manipulator dynamic equations increases the order and complexity of the overall dynamic model of the system. Accordingly, the control law required to control the system will become more complex, particularly if the controller structure is in the centralized form. Thus, due to the complexity and the structural properties of the robot manipulators dynamics, the robot manipulator has been treated as a large scale system. Consequently, the control methodologies for large scale systems, such as decomposition, decentralized control, and hierarchical control strategies, have been proposed for controlling the robot manipulators.

A procedure for decomposing and transforming the integrated nonlinear dynamic model of the robot manipulator into a set of interconnected subsystems with bounded uncertainties has been presented. The bounds on the uncertainties have been computed from the specified physical parameters of the actuators and the manipulator (length and mass of its links), the range of its payload, the joint displacements as well as the range of the velocities.

Based on a deterministic approach, a decentralized nonlinear feedback control law has been proposed for tracking control of a robot manipulator. The system is treated as a set of interconnected subsystems with bounded uncertainties. The decentralized control approach utilizes only the local states and the bounds of the uncertainties as the feedback information. Based on Lyapunov stability theory, it is shown that, with satisfaction of a certain sufficient condition, the controller results in practical stability of the overall uncertain robot manipulator system despite the absence of information signals between the local subsystems. It should be mentioned that the weaker the interconnections between the subsystems, the easier it is to satisfy the sufficient conditions. However, due to the conservative nature of the stability test, the system may be stable under the chosen control law even if the sufficient condition is not satisfied. Compared

with the linear independent joint control method, simulation results show that the method gives a better performance in terms of tracking accuracy.

In the completely decentralized control method, the interconnections between the subsystems are completely neglected, yielding a loss of information about the behaviour of the interconnections and their dynamic effects on each subsystem. Moreover, if the interconnections between the subsystem are strong, the overall system behaviour may either become unsatisfactory with respect to the accepted criterion, or unstable. To compensate the destabilizing influence of the interconnection functions, and to increase the tracking accuracy of the robot manipulator, a decentralized global control concept has been proposed. The controller utilizes the local states, the bounds on the uncertainties and the interconnection functions, as well as the states of the connecting subsystems as the feedback information. It is shown theoretically and through simulations that the error between the response of the actual robotic system and that of the reference trajectory is uniformly ultimately bounded with respect to any arbitrarily small set of ultimate boundedness; in spite of the highly nonlinear and coupled robot manipulator dynamics, and the uncertainties present in the system. It is also shown that the robot manipulator performance is superior in terms of tracking accuracy when using the decentralized global approach than when using the decentralized local approach.

As an alternative to the decentralized global control approach, two hierarchical control concepts for robot manipulators have been presented. For the first method, a linear local decentralized control law is designed for each decoupled system at the lowest level. At the upper level, a nonlinear global controller is generated to balance as exactly as possible the effects of all the nonlinearities, uncertainties, and the interconnections between the subsystems. In the second method, a nonlinear decentralized control law is designed at the lowest level, while the nonlinear global controller at the upper level is synthesized to overcome only the effects of the interconnection functions between the subsystems. All the nonlinear controllers are designed based on a deterministic approach with the assumption that all the bounds of the uncertainties, the nonlinearities, and the interconnection functions are known. It is shown theoretically and through simulations that both hierarchical controllers render the nonlinear robot manipulator practically stable and track the desired trajectory

within a particular bounded neighbourhood of the trajectory after a finite time. The controllers are found to be robust to the structural perturbations considered in the study. For the second method, it was found that the coordination signal from the upper level may be sent at a larger time interval to the lower level than the first method. This will reduce the computation of the coordination function, without significantly decreasing the tracking accuracy of the robot manipulator. Thus, the second hierarchical control concept requires less computational time. Hence, the proposed second method is more suitable for on-line control of large scale systems with fast dynamics such as robot manipulators.

Four control methods have been proposed and investigated in this study. The methods are simple and robust with regard to the uncertainties and the nonlinearities present in the system. The control laws are simple to design and implement, and require less computation time as compared to the centralized control technique. This is due to the fact that the proposed controllers are designed based on a decoupled subsystem model rather than treating the robot manipulator as a single plant. It is shown through simulation study that the performance of the proposed controllers are comparable to the centralized controller considered.

## 7.2 SUGGESTIONS FOR FUTURE RESEARCH

The integrated dynamic model of the robot manipulator derived by no means represents a complete model of the robotic system. This is because the drive system nonlinearities such as Coulomb friction, backlash, stiff spring characteristic of the actuators, and various sources of flexibility (such as, deflection of the links under load and vibrations, elastic deformation of bearings and gears) are not included in the formulation of the integrated dynamic model. Thus, further study on the performance and practicality of the proposed controller with respect to these nonlinearities and uncertainties in the formulation of a complete robot manipulator mathematical model is required.

It is assumed that the overall system input matrix for the integrated dynamic equation derived based on joint angle, velocity, and acceleration is in block diagonal form. This is true for the non-direct drive robot manipulators considered where the magnitudes of the non-zero elements of the off-block diagonal submatrices in the overall input matrix are often very small compared to the elements of the diagonal submatrices. Thus, in this case, they can be assumed negligible and can be ignored. Hence, the assumption is valid. However, if this assumption is not satisfied, such as in the case of direct drive robots in general, then further investigation on the applicability of the proposed control methods on such a system is needed.

Although the proposed control strategies perform very satisfactory during the simulation studies on a digital computer, further implementation of the control algorithms on a real industrial robot is vital in order to investigate the performance of the proposed approaches under a real situation.

### 7.3 CONCLUDING REMARKS

The major aims of this research have been achieved. It is hoped that this thesis has introduced some improvements in the areas of modelling, decentralized and hierarchical control of a fast and strongly nonlinear system, such as a robot manipulator, that will lead to further scientific and commercial exploitation.

## APPENDIX A

### DENAVIT - HARTENBERG (D-H) NOTATION AND TRANSFORMATION MATRIX

An algorithm for establishing the coordinate frames and Denavit-Hartenberg (D-H) transformation matrices for robot manipulators is presented in this appendix.

It is assumed that an  $N$  dof robot manipulator has  $N+1$  links ( $N$  moving links and one base link) numbered from 0 to  $N$  starting from the base link, which is taken as link 0. The joints are numbered from 1 to  $N$ , so that the  $i$ th joint couples links  $i-1$  and  $i$ .

The translational and rotational relationship of each link relative to its neighbours can be described by using the D-H convention [Craig, 1986; Shahinpoor, 1987; Spong and Vidyasagar, 1989] for each link of the robot manipulator. A right-handed coordinate frame is assigned to each link and normally, for link  $i$ , the coordinate frame is at joint  $i+1$ . The  $z_i$ -axis is the axis of revolution (translation) of joint  $i+1$  if joint  $i+1$  is revolute (prismatic). The  $x_i$ -axis is in the direction normal to both the  $z_i$  and  $z_{i-1}$  axes, pointing away from the  $z_{i-1}$ -axis towards the  $z_i$ -axis. The axis  $y_i$  is chosen to complete the right-handed coordinate system. The D-H parameters ( $a_i$ ,  $d_i$ ,  $\alpha_i$ , and  $\theta_i$ ) are then obtained for each link. For a revolute joint,  $a_i$ ,  $d_i$ , and  $\alpha_i$  are constant while  $\theta_i$  is the joint variable. For a prismatic joint,  $a_i$ ,  $\theta_i$ , and  $\alpha_i$  are constant, while  $d_i$  varies (the joint variable). When the joint axes of adjacent joints intersect, the link length  $a_i$  is zero, such as the case with prismatic joints. Figure A-1 illustrate the D-H frame assignment.

Once the link coordinate systems have been established for the robot manipulator, the homogeneous D-H transformation matrix  $A_{i-1}^i$ , relating the  $i$ th

coordinate frame to the  $i$ -1th coordinate frame can be obtained.

The method may be summarized by the following algorithm :

Step 1 : Establish the base frame. The  $z_o$ -axis lies along the axis of motion or rotation of the joint one. The  $x_o$  and  $y_o$  axes are chosen conveniently to form a right-hand frame.

Step 2 : For each  $i$ ,  $i = 1, 2, \dots, N-1$ , do Steps 3 to 6.

Step 3 : Establish the  $z_i$ -axis. Align the  $z_i$ -axis with the axis of motion (rotary or sliding) of joint  $i+1$ .

Step 4 : Locate the origin of the  $i$ th coordinate frame  $o_i$  at the intersection of the common normal between the  $z_i$  and  $z_{i-1}$ -axes and the  $z_i$ -axis. If  $z_i$ -axis intersects  $z_{i-1}$ -axis, locate  $o_i$  at this intersection.

Step 5 : Establish  $x_i$ -axis along the common normal between  $z_{i-1}$  and  $z_i$  through  $o_i$ , or in the direction normal to the  $z_{i-1} - z_i$  plane if  $z_{i-1}$  and  $z_i$  intersect.

Step 6 : Establish  $y_i$  to complete a right-handed coordinate system.

Step 7 : Establish the end-effector frame  $o_N x_N y_N z_N$ . Assuming the  $N$ th joint is revolute, set  $z_N$  along the direction of approach of the end-effector on an object. Establish the origin  $o_N$  conveniently along  $z_N$ , preferably at the center of the gripper or at the tip of any tool that the robot manipulator may be carrying. Set  $y_N$  in the direction of the gripper closure, and set  $x_N$  to complete a right-hand frame. If the tool is not a simple gripper, set  $x_N$  and  $y_N$  conveniently to form a right-handed frame.

Step 8 : Create a table of the link parameters (D-H parameters)  $a_i, d_i, \alpha_i, \theta_i$ :



$a_i$  = the common normal distance along  $x_i$ -axis from the intersection of the  $x_i$  and  $z_{i-1}$ -axes to the origin of the  $i$ th coordinate system,  $o_i$ . Note that  $a_i$  is also referred to as the length of the  $i$ th link.

$d_i$  = the distance along  $z_{i-1}$  from the origin of the  $i-1$ th coordinate system,  $o_{i-1}$ , to the intersection of the  $x_i$  and  $z_{i-1}$ -axes. (The distance between the two common normals  $a_i$  and  $a_{i-1}$ , or the distance between  $x_i$  and  $x_{i-1}$  axes.) Note that  $d_i$  is the joint variable if joint  $i$  is prismatic.

$\alpha_i$  = the twist angle from the  $z_{i-1}$ -axis to the  $z_i$ -axis, measured about the  $x_i$ -axis. For most commercial manipulator, the twist angles are in the multiples of  $90^\circ$ .

$\theta_i$  = the angle of rotation from the  $x_{i-1}$ -axis to the  $x_i$ -axis, measured about the  $z_{i-1}$ -axis. Note that  $\theta_i$  is the joint variable if joint  $i$  is revolute.

Step 9 : Form the D-H transformation matrices  $A_{i-1}^i$  by substituting the D-H parameters into the following equation :

$$A_{i-1}^i = \begin{bmatrix} C_{\theta_i} & -S_{\theta_i}C_{\alpha_i} & S_{\theta_i}S_{\alpha_i} & a_iC_{\theta_i} \\ S_{\theta_i} & C_{\theta_i}C_{\alpha_i} & -C_{\theta_i}S_{\alpha_i} & a_iS_{\theta_i} \\ 0 & S_{\alpha_i} & C_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (A.1)$$

where  $S_{\theta_i}$  and  $C_{\theta_i}$  are  $\sin(\theta_i)$  and  $\cos(\theta_i)$  respectively.

Once the D-H transformation matrices, where each matrix relates the consecutive frames of the robot manipulator have been established, the position and orientation of the tool (end-effector) frame at the outermost link can be expressed in terms of the coordinate system established at the based of the robot

manipulator, that is, the base coordinate frame. For an  $N$  dof robot manipulator, the expression for the end-effector coordinate frame is given by

$$A_0^N = A_0^1 A_1^2 A_2^3 \dots A_{N-1}^N. \quad (A.1)$$

Since each  $A_{i-1}^i$  matrix is a function of an appropriate joint variable of the corresponding link, the  $A_0^N$  matrix will be a function of all the joint variables of the robot manipulator.

Another important application of the D-H transformation matrix is that the D-H matrices can be coupled to the Euler-Lagrange equation to derive the dynamic model of the mechanical linkage of the robot manipulator systematically as outlined in Appendix B.

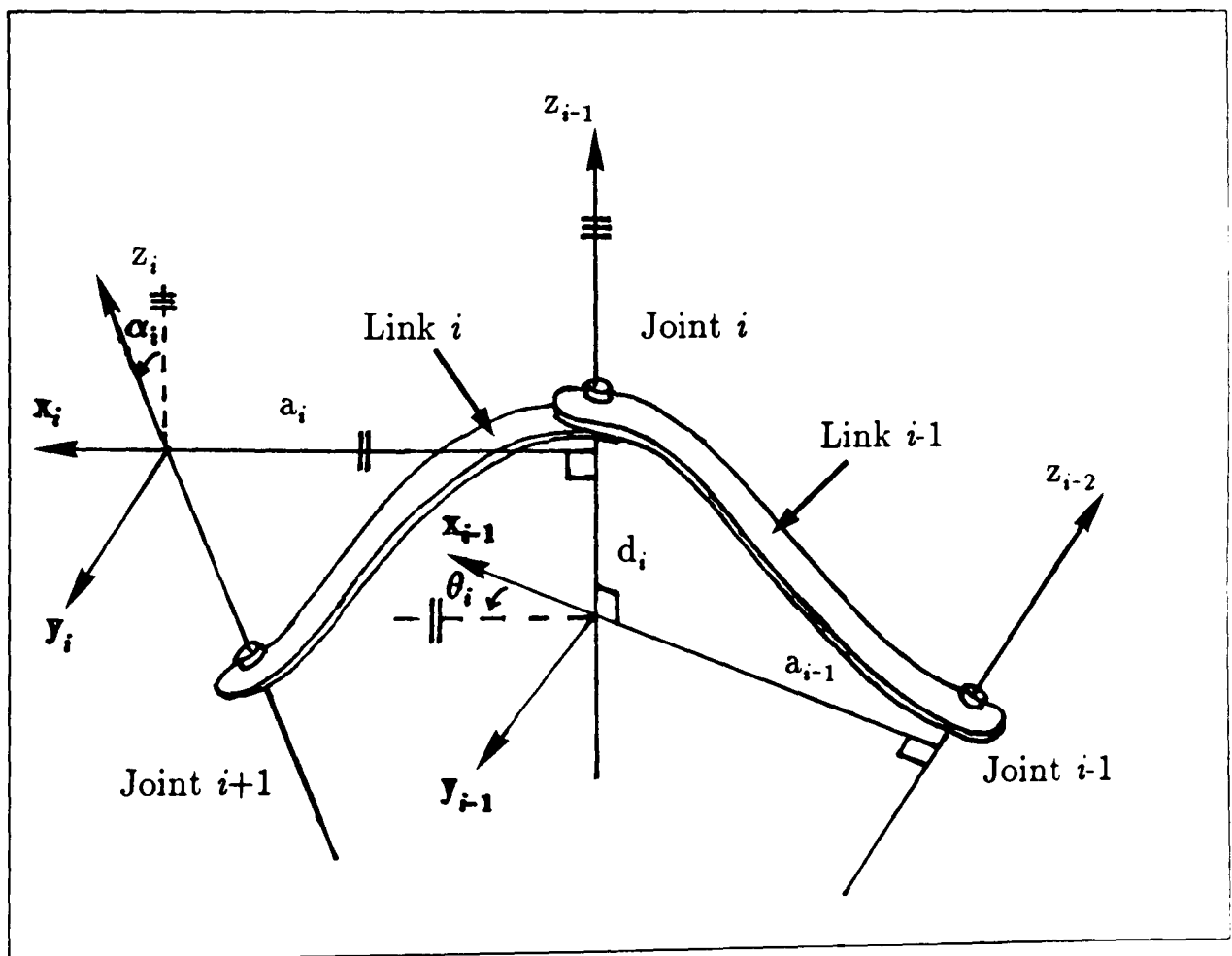


FIGURE A-1 : Denavit-Hartenberg Frame Assignment [Spong and Vidyasagar, 1981]

## APPENDIX B

### EULER-LAGRANGE FORMULATION OF MANIPULATOR LINKS DYNAMICS

The application of the Denavit-Hartenberg (D-H) matrix representation of the robot manipulator linkage and the Euler-Lagrange equation results in a systematic, convenient, and compact formulation of the robot manipulator dynamic equations. In the following, the method [Lee, 1983; Paul, 1981] is briefly presented.

The Lagrange function  $L(\theta, \dot{\theta})$  for an  $N$  dof robot manipulator can be obtained as follows :

$$L(\theta, \dot{\theta}) = \mathfrak{K}(\theta, \dot{\theta}) - \mathfrak{P}(\theta) , \quad (\text{B.1})$$

$\mathfrak{K}(\theta, \dot{\theta})$  : Total kinetic energy of the system

$\mathfrak{P}(\theta)$  : Total potential energy of the system

$\theta$  : Generalized coordinate of the system

$\theta \in \mathbb{R}^N$  .

For  $N$  dof robot manipulator, the kinetic and potential energy are given as follows :

$$\mathfrak{K}(\theta, \dot{\theta}) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^N \text{Tr} [\mathbf{U}_{ij} \mathbf{J}_i \mathbf{U}_{ik}^T] \dot{\theta}_j \dot{\theta}_k , \quad (\text{B.2})$$

$$\mathfrak{P}(\theta) = - \sum_{i=1}^N m_i \bar{g} \mathbf{A}_o^i \bar{\mathbf{T}}_i , \quad (\text{B.3})$$

where

$$\mathbf{U}_{ij} = \begin{cases} \frac{\partial \mathbf{A}_o^i}{\partial \theta_j} = \mathbf{A}_o^{j-1} \mathbf{Q}_j \mathbf{A}_{j-1}^j & : \text{ for } j \leq i \\ 0 & : \text{ for } j > i \end{cases} \quad (\text{B.4})$$

$\mathbf{J}_i$  = pseudo inertia matrix of the  $i$ th link

$m_i$  = mass of the  $i$ th link

$\text{Tr}$  = trace operator

$\bar{\mathbf{g}}$  = gravity row vector

$\bar{\mathbf{r}}_i$  = the position vector of the centre of gravity of the  $i$ th link with respect to the  $i$ th coordinate frame.

$\mathbf{A}_{i-1}^i$  = 4x4 homogeneous D-H transformation matrix that relates the  $i$ th coordinate frame to the  $i-1$ th coordinate system (see Appendix A).

Then, from the Euler-Lagrange equation of motion ,

$$\frac{d}{dt} \left[ \frac{\partial L(\theta, \dot{\theta})}{\partial \dot{\theta}_i} \right] - \left[ \frac{\partial L(\theta, \dot{\theta})}{\partial \theta_i} \right] = T_i, \quad i \in \mathcal{J} \quad (\text{B.5})$$

$T_i$  = generalized torque or force on the  $i$ th joint,

the dynamic equations of the  $N$  dof robot manipulator can be obtained as follows :

$$\mathbf{T} = \mathbf{M}(\theta) \ddot{\theta} + \mathbf{D}(\theta, \dot{\theta}) + \mathbf{G}(\theta), \quad (\text{B.6})$$

where,

$$\mathbf{M}_{ik} = \sum_{j=\max(i,k)}^N \text{Tr} \left[ \mathbf{U}_{jk} \mathbf{J}_j \mathbf{U}_{ji}^T \right] \quad (\text{B.7})$$

$$D_i = \sum_{k=1}^N \sum_{m=1}^N D_{ikm} \dot{\theta}_k \dot{\theta}_m \quad (B.8)$$

$$D_{ikm} = \sum_{j=\max(i,k,m)}^N \text{Tr} [U_{jkm} J_j U_{ji}^T] \quad (B.9)$$

$$G_i = \sum_{j=i}^N [-m_j \bar{g} U_{ji} \bar{r}_j] \quad (B.10)$$

$$U_{ijk} \triangleq \frac{\partial U_{ij}}{\partial \theta_k} = \begin{cases} A_o^{j-1} Q_j A_{j-1}^{k-1} Q_k A_{k-1}^i, & i \geq k \geq j \\ A_o^{k-1} Q_k A_{k-1}^{j-1} Q_j A_{j-1}^i, & i \geq j \geq k \\ 0, & \text{otherwise} \end{cases} \quad (B.11)$$

$$\bar{g} = \begin{bmatrix} 0 & 0 & -g & 0 \end{bmatrix}^T \quad (B.12)$$

$$Q_i = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{for a revolute joint} \quad (B.13)$$

$$Q_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{for a prismatic joint}$$

$\mathbb{J}_i$  = pseudo inertia matrix of the  $i$ th link, where

$$\mathbb{J}_i = \begin{bmatrix} \frac{-\mathbb{I}_{xx}^i + \mathbb{I}_{yy}^i + \mathbb{I}_{zz}^i}{2} + m_i \bar{x}_i^2 & \mathbb{I}_{xy}^i & \mathbb{I}_{xz}^i & m_i \bar{x}_i \\ \mathbb{I}_{xy}^i & \frac{\mathbb{I}_{xx}^i - \mathbb{I}_{yy}^i + \mathbb{I}_{zz}^i}{2} + m_i \bar{y}_i^2 & \mathbb{I}_{yz}^i & m_i \bar{y}_i \\ \mathbb{I}_{xy}^i & \mathbb{I}_{yz}^i & \frac{\mathbb{I}_{xx}^i + \mathbb{I}_{yy}^i - \mathbb{I}_{zz}^i}{2} + m_i \bar{z}_i^2 & m_i \bar{z}_i \\ m_i \bar{x}_i & m_i \bar{y}_i & m_i \bar{z}_i & m_i \end{bmatrix}$$

or

(B.14)

$$\mathbb{J}_i = m_i \begin{bmatrix} \frac{-K_{ixx}^2 + K_{iyy}^2 + K_{izz}^2}{2} & K_{ixy}^2 & K_{ixz}^2 & \bar{x}_i \\ K_{ixy}^2 & \frac{K_{ixx}^2 - K_{iyy}^2 + K_{izz}^2}{2} & K_{iyz}^2 & \bar{y}_i \\ K_{ixy}^2 & K_{iyz}^2 & \frac{K_{ixx}^2 + K_{iyy}^2 - K_{izz}^2}{2} & \bar{z}_i \\ \bar{x}_i & \bar{y}_i & \bar{z}_i & 1 \end{bmatrix}$$

(B.15)

and,

- $K_{ijk}$  : the radius of the  $i$ th link gyrating about the  $j-k$  axes
- $\bar{x}_i, \bar{y}_i, \bar{z}_i$  : coordinates of the centre of gravity of the  $i$ th link
- $\mathbb{I}_{xx}, \mathbb{I}_{yy}, \mathbb{I}_{zz}$  : moments of inertia about axes situated at the joint.

The Coriolis and centrifugal vector  $D(\theta, \dot{\theta})$  can also be written in the following form [Lim and Eslami, 1985] :

$$D(\theta, \dot{\theta}) = \bar{D}(\theta) V(\dot{\theta}) \dot{\theta}(t) , \quad (B.16)$$

where

$$V(\dot{\theta}) = \begin{bmatrix} \dot{\theta}_1 I_N \\ O_{N-1,1} & \dot{\theta}_2 I_{N-1} \\ \vdots & \vdots \\ O_{1,N-1} & \dot{\theta}_N \end{bmatrix} , \quad (B.17)$$

where  $\bar{D}(\theta)$  is a  $N \times \sum_{i=1}^N i$  matrix,  $V(\dot{\theta})$  is a  $\sum_{i=1}^N i \times N$  matrix,  $O_{i,j}$  is an  $i \times j$  null matrix, and  $I_k$  is a  $k \times k$  identity matrix. Hence equation (B.6) can be rewritten as

$$T = M(\theta) \ddot{\theta} + \bar{D}(\theta) V(\dot{\theta}) \dot{\theta}(t) + G(\theta) \quad (B.18)$$

In some cases, it is preferable to rewrite the dynamic equation of the manipulator linkage in the following form :

$$T = M(\theta) \ddot{\theta} + \hat{D}(\theta, \dot{\theta}) \dot{\theta} + \hat{G}(\theta) \theta , \quad (B.19)$$

where  $\hat{D}(\theta, \dot{\theta})$  and  $\hat{G}(\theta)$  are  $N \times N$  matrices given as follow :

$$\hat{D}(\theta, \dot{\theta}) = \bar{D}(\theta) V(\dot{\theta}) , \quad (B.20)$$

and the matrix  $\hat{G}(\theta) \in \mathbb{R}^{N \times N}$  which is not unique [ Lim and Eslami, 1985; 1986], can be obtained from

$$G(\theta) = \hat{G}(\theta) \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix} . \quad (B.21)$$

# APPENDIX C

## ROBOT MANIPULATOR COMPLETE MODEL - A SURVEY

In this appendix, the complete model of the robot manipulator as given by Vukobratovic and Potkonjak [1982], Vukobratovic et.al. [1985], Troch [1986], and Troch *et.al.* [1986] are presented for comparison purposes.

In the following, the manipulator link dynamics (equation 1.1) and the actuator dynamics (equation 2.23) are reintroduced for convenience. The dynamic model of the mechanical links of an  $N$  dof robot manipulator is as follows :

$$M(\theta(t), \xi) \ddot{\theta}(t) + D(\theta(t), \dot{\theta}(t), \xi) + G(\theta(t), \xi) = T(t) , \quad (C.1)$$

$$\theta(t) = [ \theta_1(t), \theta_2(t), \dots, \theta_N(t) ]^T$$

$$\theta(t) \in \mathbb{R}^N , \quad \dot{\theta}(t) \in \mathbb{R}^N , \quad \ddot{\theta}(t) \in \mathbb{R}^N .$$

For  $N$  actuators ( $N$  dof robot manipulator), the augmented dynamic equation of the actuators can be written in compact form as follows :

$$\dot{X}_A(t) = A_A X_A(t) + B_A U(t) + F_A T(t) , \quad (C.2)$$

$$X_A(t) = [ X_{A1}^T(t), X_{A2}^T(t), \dots, X_{AN}^T(t) ]^T$$

$$X_{Ai}^T(t) = [ \theta_i(t), \dot{\theta}_i(t) , i_{ai}(t) ] \quad (C.3)$$





$$A_A(X_A) = A_A X_A + F_A [I_N - M(X_A) Z_C F_A]^{-1} \left\{ M(X_A) Z_C A_A X_A + D(X_A) + G(X_A) \right\} \quad (C.9)$$

$$B_A(X_A) = B_A + F_A [I_N - M(X_A) Z_C F_A]^{-1} M(X_A) Z_C B_A . \quad (C.10)$$

**C2. Method Of Vukobratovic et.al. [1985], Troch [1986],  
And Troch et.al. [1986] :**

By substituting equation (C.6) into the actuators state equation (C.2), and after a simple manipulation, the state equation of the robot manipulator consisting the actuators as well as the mechanical links dynamics is obtained as follows :

$$\dot{X}_A(t) = A(X_A) + B(X_A) U(t) , \quad (C.11)$$

where

$$A(X_A) = [ I_{3N} - F_A M(X_A) Z_C ]^{-1} \left\{ A_A X_A + F_A [D(X_A) + G(X_A)] \right\} \quad (C.12)$$

$$B(X_A) = [ I_{3N} - F_A M(X_A) Z_C ]^{-1} B_A . \quad (C.13)$$

Clearly, the method of deriving the integrated model of the robot manipulator presented in this section is much simpler than the method outlined in section C1 above. In the following, it will be shown that the state equation of the integrated model of the robot manipulator presented in section 2.4.1 of Chapter 2 (equations 2.35, 2.36, and 2.37), is equivalent to the state equation of the robot manipulator presented in this section (equations C.11, C.12 and C.13).

From Chapter 2, since  $B_p^\dagger(X_A) = [ O_{NN} : M(X_A) ]$ , and by the fact that the  $2N \times 3N$  transformation matrix  $Z_A$  has the following structure :

$$\mathbf{Z}_A = \begin{bmatrix} 1 & 0 & 0 & & & \\ - & - & - & - & - & \\ & & 1 & 0 & 0 & \mathbf{0} \\ & & - & - & - & \\ & \mathbf{0} & & & & - & - & - & - \\ & & & & & & 1 & 0 & 0 \\ \dots & & & & & & & & \\ & & & & & & & & \mathbf{Z}_C \end{bmatrix}, \quad (\text{C.14})$$

by a simple mathematical manipulation, it can be shown that

$$\mathbf{I}_{3N} - \mathbf{F}_A \mathbf{B}_p^\dagger(\mathbf{X}_A) \mathbf{Z}_A = \mathbf{I}_{3N} - \mathbf{F}_A \mathbf{M}(\mathbf{X}_A) \mathbf{Z}_C, \quad (\text{C.15})$$

and

$$\left[ \mathbf{A}_A - \mathbf{F}_A \mathbf{B}_p^\dagger(\mathbf{X}_A) \mathbf{A}_p(\mathbf{X}_A) \mathbf{Z}_A \right] \mathbf{X}_A = \mathbf{A}_A \mathbf{X}_A + \mathbf{F}_A [\mathbf{D}(\mathbf{X}_A) + \mathbf{G}(\mathbf{X}_A)], \quad (\text{C.16})$$

then equation (2.35) in Chapter 2 can be rewritten as

$$\dot{\mathbf{X}}_A(t) = \left[ \mathbf{I}_{3N} - \mathbf{F}_A \mathbf{M}(\mathbf{X}_A) \mathbf{Z}_C \right]^{-1} \left\{ \mathbf{A}_A \mathbf{X}_A + \right. \\ \left. \mathbf{F}_A [\mathbf{D}(\mathbf{X}_A) + \mathbf{G}(\mathbf{X}_A)] + \mathbf{B}_A \mathbf{U}(t) \right\}, \quad (\text{C.17})$$

which is similar to the integrated model (equations C.11 - C.14) presented in section C2 above.

## APPENDIX D

### DETAILED EXPRESSIONS OF THE INTEGRATED DYNAMIC MODEL OF THE THREE DOF REVOLUTE ROBOT MANIPULATOR

In this appendix, the expression for the non-zero nonlinear elements of the integrated dynamic model of the three dof robot manipulator as derived in Chapter 2, that is equations (2.75), (2.76), and (2.77), are presented.

Let,

$$\begin{aligned}\varphi_1 &= 1 - w_{B1}M_{11} \\ \varphi_2 &= 1 - w_{B2}M_{22} \\ \varphi_3 &= 1 - w_{B3}M_{33} \\ \varphi_4 &= -w_{B2}M_{23} \\ \varphi_5 &= -w_{B3}M_{23} \\ \Psi &= \varphi_2\varphi_3 - \varphi_4\varphi_5.\end{aligned}\tag{D.1}$$

Then, the non-zero nonlinear elements of the system matrix  $A(X_B, t)$  can be obtained as follows :

$$a_{32} = \left\{ a_{B32} \right\} / \varphi_1$$

$$\begin{aligned}
a_{33} &= \left\{ a_{B33} + f_{B1} M_{11} + w_{B1} \tilde{C}_{11} \right\} / \varphi_1 \\
a_{35} &= \left\{ f_{B1} \bar{D}_{12} x_2 + w_{B1} \tilde{D}_{12} \right\} / \varphi_1 \\
a_{36} &= \left\{ w_{B1} \tilde{C}_{12} \right\} / \varphi_1 \\
a_{38} &= \left\{ f_{B1} \bar{D}_{13} x_2 + w_{B1} \tilde{D}_{13} \right\} / \varphi_1 \\
a_{39} &= \left\{ w_{B1} \tilde{C}_{13} \right\} / \varphi_1 \\
a_{62} &= \left\{ \varphi_3 [ f_{B2} \bar{D}_{23} x_2 + w_{B2} \tilde{D}_{21} ] - \varphi_4 [ f_{B3} \bar{D}_{31} x_2 + w_{B3} \tilde{D}_{31} ] \right\} / \Psi \\
a_{63} &= \left\{ \varphi_3 w_{B2} \tilde{C}_{21} - \varphi_4 w_{B3} \tilde{C}_{31} \right\} / \Psi \\
a_{64} &= \left\{ \varphi_3 f_{B2} G_2 \right\} / \left\{ \Psi x_4 \right\} \\
a_{65} &= \left\{ \varphi_3 [ a_{B65} + w_{B2} \tilde{D}_{22} ] - \varphi_4 [ f_{B3} \bar{D}_{34} x_5 + w_{B3} \tilde{D}_{32} ] \right\} / \Psi \\
a_{66} &= \left\{ \varphi_3 [ a_{B66} + f_{B2} M_{22} + w_{B2} \tilde{C}_{22} ] - \varphi_4 [ f_{B3} M_{23} + w_{B3} \tilde{C}_{32} ] \right\} / \Psi \\
a_{67} &= - \left\{ \varphi_4 f_{B3} G_3 \right\} / \left\{ \Psi x_7 \right\} \tag{D.2} \\
a_{68} &= \left\{ \varphi_3 [ f_{B2} ( \bar{D}_{25} x_5 + \bar{D}_{26} x_8 ) + w_{B2} \tilde{D}_{23} ] - \varphi_4 [ a_{B98} + w_{B3} \tilde{D}_{33} ] \right\} / \Psi \\
a_{69} &= \left\{ \varphi_3 [ f_{B2} M_{23} + w_{B2} \tilde{C}_{23} ] - \varphi_4 [ a_{B99} + f_{B3} M_{33} ] \right\} / \Psi \\
a_{92} &= \left\{ \varphi_2 [ f_{B3} \bar{D}_{31} x_2 + w_{B3} \tilde{D}_{31} ] - \varphi_5 [ f_{B2} \bar{D}_{21} x_2 + w_{B2} \tilde{D}_{21} ] \right\} / \Psi
\end{aligned}$$

$$a_{93} = \left\{ \varphi_2 w_{B3} \tilde{C}_{31} - \varphi_5 w_{B2} \tilde{C}_{21} \right\} / \Psi$$

$$a_{94} = - \left\{ \varphi_5 f_{B2} G_2 \right\} / \left\{ \Psi x_4 \right\}$$

$$a_{95} = - \left\{ \varphi_5 [a_{B65} + w_{B2} \tilde{D}_{22}] - \varphi_2 [f_{B3} \bar{D}_{34} x_5 + w_{B3} \tilde{D}_{32}] \right\} / \Psi$$

$$a_{96} = - \left\{ \varphi_5 [a_{B66} + f_{B2} M_{22} + w_{B2} \tilde{C}_{22}] - \varphi_2 [f_{B3} M_{23} + w_{B3} \tilde{C}_{32}] \right\} / \Psi$$

$$a_{97} = \left\{ \varphi_2 f_{B3} G_3 \right\} / \left\{ \Psi x_7 \right\}$$

$$a_{98} = - \left\{ \varphi_5 [f_{B2} (\bar{D}_{25} x_5 + \bar{D}_{26} x_8) + w_{B2} \tilde{D}_{23}] - \varphi_2 [a_{B98} + w_{B3} \tilde{D}_{33}] \right\} / \Psi$$

$$a_{99} = - \left\{ \varphi_5 [f_{B2} M_{23} + w_{B2} \tilde{C}_{23}] - \varphi_2 [a_{B99} + f_{B3} M_{33}] \right\} / \Psi .$$

For the input matrix  $B(X_B, t)$ , its non-zero elements are as follows :

$$b_{11} = \left\{ b_{B1} \right\} / \varphi_1$$

$$b_{22} = \left\{ b_{B2} \varphi_3 \right\} / \Psi$$

$$b_{23} = - \left\{ b_{B3} \varphi_4 \right\} / \Psi \tag{D.3}$$

$$b_{32} = - \left\{ b_{B2} \varphi_5 \right\} / \Psi$$

$$b_{33} = \left\{ b_{B3} \varphi_2 \right\} / \Psi .$$

## REFERENCES

- Ailon, A., and Langholz, G., [1988] : A Study Of Controllability And Time-Optimal Control Of A Robot Model With Drive Train Compliances And Actuator Dynamics. *IEEE Trans. Auto. Control*, AC-33, pp. 852-856.
- Ambrosino, G., Celentano, G., and Garofalo, F., [1988] : Adaptive Tracking Control Of Industrial Robots. *ASME J. Dynam. Syst. Meas. and Control*, 110, pp. 215-220.
- Balestrino, A., De Maria, G., and Sciavicco, L., [1983] : An Adaptive Model Following Control For Robotic Manipulators. *ASME J. Dynam. Syst. Meas. and Control*, 105, pp. 143-151.
- Beekmann, R. W., and Lee, K. Y., [1988] : Nonlinear Robotic Control Including Drive Motor Interactions. *Proc. American Contr. Conf.*, Atlanta, USA, pp. 1333-1338.
- Bejczy, A. K., [1974] : Robot Arm Dynamics And Control. *Jet Propulsion Lab. Tech. Memo. 33-669*, Pasadena, California, USA.
- Bestaoui, Y., [1988] : Adaptive Hierarchical Control For Robotic Manipulators. *Robotics and Autonomous Systems*, 4, pp. 145-155.
- Chen, Y. F., Mita, T., and Wakui, S., [1990] : A New And Simple Algorithm For Sliding Mode Trajectory Control Of Robot Arm. *IEEE Trans. Auto. Control*, AC-35, pp. 828-829.
- Chen, Y. H., [1986] : On The Deterministic Performance Of Uncertain Dynamical Systems. *Int. J. Control*, 43, pp. 1557-1579.

- Chen, Y. H., [1987a] : Deterministic Control Of Large-Scale Uncertain Dynamical Systems. *J. Franklin Inst.*, **323**, pp.135-144.
- Chen, Y. H., [1987b] : Adaptive Robust Model-Following Control And Application To Robot Manipulators. *ASME J. Dynam. Syst. Meas. and Control*, **109**, pp 209-215.
- Chen, Y. H., [1988] : Decentralized Robust Control System Design For Large-Scale Uncertain Systems. *Int. J. Control*, **47**, pp. 1195-1205.
- Chen, Y. H., and Eyo, V. A., [1988] : Robust Computed Torque Control Of Mechanical Manipulators : Non-Adaptive vs. Adaptive. *Proc. American Contr. Conf.*, Atlanta, USA, pp. 1327-1332.
- Chen, Y. H., and Pandey, S., [1990] : Uncertain Bound-Based Hybrid Control For Robot Manipulators. *IEEE Trans Robotics And Automation*, **RA-6**, pp. 303-311.
- Corless, M. J., [1989] : Tracking Controllers for Uncertain Systems : Application To A Manutec R3 Robot. *ASME J. Dynam. Syst. Meas. and Control*, **111**, pp. 609-618.
- Corless, M. J., and Leitmann, G., [1981] : Continuous State Feedback Guaranteeing Uniform Ultimate Boundedness For Uncertain Dynamical Systems. *IEEE Trans. Auto. Control*, **AC-26**, pp. 1139-1144.
- Corless, M. J., Leitmann, G., and Ryan, E. P., [1984] : Tracking In The Presence Of Bounded Uncertainties. *Proc. of the Fourth IMA Int. Conf. on Control Theory*, Cambridge, Cook, P. A. (ed), Academic Press, pp. 347-361.
- Craig, J. J., [1986] : *Introduction To Robotics, Mechanics and Control*. Addison-Wesley Publ. Comp., USA.



- Davison, E. J., and Özgüner, Ü., [1983] : Characterizations Of Decentralized Fixed Modes For Interconnected Systems. *Automatica*, 19, pp. 169-182.
- Dubowsky, S., and DesForges, D., [1979] : The Application Of Model Referenced Adaptive Control To Robotic Manipulators. *ASME J. Dynam. Syst. Meas. and Control*, 101, pp. 193-200.
- Franklin, J. N., [1968] : *Matrix Theory*. Prentice-Hall, Englewood Cliffs, New Jersey, USA.
- Freund, E., [1982] : Fast Nonlinear Control With Arbitrary Pole-Placement For Industrial Robots And Manipulators. *Int. J. Robotic Research*, 1, pp. 65-78.
- Fu, K. S., Gonzalez, R. C., and Lee, C. S. G., [1987] ; *Robotics : Control, Sensing, Vision, And Intelligence*. McGraw-Hill Int. Editions, Singapore.
- Fu, L. C., and Liao, T. L., [1990] ; Globally Stable Robust Tracking Of Nonlinear Systems Using Variable Structure Control And With An Application To A Robotic Manipulator. *IEEE Trans. Auto. Control*, AC-35, pp. 1345-1350.
- Good, M. C., Sweet, L. M., and Strobel, K. L., [1985] : Dynamic Models For Control System Design Of Integrated Robot And Drive Systems. *ASME J. Dynam. Syst. Meas. and Control*, 107, pp. 53-59.
- Gutman, S., and Palmor, Z., [1982] : Properties of Min-Max Controllers In Uncertain Dynamical Systems. *SIAM J. Control and Optim.*, 20, pp. 850-861.
- Hollerbach, J. M., [1980] : A Recursive Formulation Of Lagrange Manipulator Dynamics. *IEEE Trans. Syst., Man, and Cybern.*, SMC-10, pp. 730-736.
- James, M. L., Smith, G. M., and Wolford, J. C., [1985] : *Applied Numerical Methods For Digital Computation*. Third Edition. Harper and Row Publishers, New York, New York.

- Katic, D., and Vukobratovic, M. K., [1986] : The Influence Of Actuator Model Complexity On Control Synthesis For High Performance Robot Trajectory Tracking. *Theory of Robots : Selected Papers From The IFAC/IFIP/IMACS Symp.*, Vienna, Austria, Kopecek, P., Troch, I., and Desoyer, K. (eds), pp. 217-222.
- Koivo, A., and Guo, T., [1983] : Adaptive Linear Controller For Robotic Manipulators. *IEEE Trans. Auto. Control*, AC-28, pp. 162-171.
- Kuo, C. Y., and Wang, S. P. T., [1989] : Nonlinear Robust Industrial Robot Control. *ASME J. Dynam. Syst. Meas. and Control*, 111, pp. 24-30.
- Lee, C. S. G., [1982] : Robot Arm Kinematics, Dynamics, And Control. *Computer*, 15, pp. 62-80.
- Lee, C. S. G., [1983] : Robot Arm Dynamics. *Tutorial On Robotics*, Lee, C. S. G., Gonzalez, R. C., and Fu, K. S. (eds), IEEE Computer Society Press, Silver Spring, USA, pp. 93-102.
- Leininger, G. G., [1984] : Adaptive Control Of Manipulators Using Self-Tuning Methods. *Robotic Research : The First Int. Symp.*, Brady, M., and Paul, R. P. (eds), The MIT Press, Cambridge, Massachusetts.
- Leitmann, G., [1979] : Guaranteed Asymptotic Stability For Some Linear Systems With Bounded Uncertainties. *ASME J. Dynam. Syst. Meas. and Control*, 101, pp. 212-216.
- Leitmann, G., [1981] : On The Efficacy Of Nonlinear Control In Uncertain Linear Systems. *ASME J. Dynam. Syst. Meas. and Control*, 102, pp. 95-102.
- Lim, K. Y., and Eslami, M., [1985] : Adaptive Controller Designs For Robot Manipulator Systems Using Lyapunov Direct Method. *IEEE Trans. Auto. Control*, AC-30, pp. 1229-1233.

- Lim, K. Y., and Eslami, M., [1986] : Robust Adaptive Controller Designs For Robot Manipulator Systems. *Proc. IEEE Int. Conf. On Robotics And Automation*, San Francisco, California, USA, pp. 1209-1215.
- Luh, J. Y. S., Walker, M. W., and Paul, R. P., [1980a] : Resolved-Acceleration Control Of Mechanical Manipulators. *IEEE Trans. Auto. Control*, AC-25, pp. 468-474.
- Luh, J. Y. S., Walker, M. W., and Paul, R. P., [1980b] : On-line Computational Scheme for Mechanical Manipulators. *ASME J. Dynam. Syst. Meas. and Control*, 102, pp. 69-76.
- Markiewicz, B. R., [1973] : Analysis Of The Computed Torque Drive Method And Comparison With Conventional Position Servo For A Computer Controlled Manipulator. *Jet Propulsion Lab. Tech. Memo. 33-601*, Pasadena, California, USA.
- Michel, A. N., and Miller, R. K., [1977] : *Qualitative Analysis Of Large Scale Dynamical Systems*. Academic Press Inc., New York.
- Mikhailov, L. K., Zaprianov, J. D., and Naplatanov, N. D., [1985] : Hierarchical Approach To Robot Control System Synthesis. *Robot Control (SYROCO'85) : Proc First IFAC Symp.*, Barcelona, Spain, Basanez, L., Ferrate, G., and Saridis, G. N. (eds), pp. 163-167.
- Mills, J. K., and Goldenberg, A. A., [1988] : Global Connective Stability Of A Class Of Robotic Manipulators. *Automatica*, 24, pp. 835-839.
- Nicosia, S., and Tomei, P. [1984] : Model Reference Adaptive Control Algorithm For Industrial Robot. *Automatica*, 20, pp. 635-644.
- Orin, D. E., McGhee, R. B., Vukobratovic, M. K., and Hartoch, G., [1979] : Kinematic And Kinetic Analysis Of Open-Chain Linkages Utilizing Newton-Euler Methods. *Mathematical Biosciences*, 43, pp. 107-130.

- Osman, J. H. S., [1985] : *Design Of Variable Structure Adaptive Model Following Controller For Robot Manipulators*. MSc. Thesis, Southern Illinois Univ., Carbondale, Illinois, USA.
- Pandian, S. R., Hanmandlu, M., and Gopal, M., [1988] : A Decentralized High Gain Controller For Trajectory Control Of Robot Manipulators. *Robotic Control, Theory and Applications*, Warwick, K., and Pugh, A. (eds), Peter Peregrinus Ltd, London, pp. 90-100.
- Paul, R. P., [1981] : *Robot Manipulators : Mathematics, Programming, And Control*. MIT Press, Cambridge, Massachusetts, USA.
- Pollard, S. B., Porrill, J., Mayhew, J. E. W., and Frisby, J. P., [1986] : Disparity Gradient, Lipschitz Continuity, and Computing Binocular Correspondences. in *Robotic Research, The third Int. Symp.*, Fougères, O. D., and Giralt, G. (eds). The MIT Press, Cambridge, Massachusetts, USA., pp. 19-26.
- Raibert, M. H., and Horn, B. K. P., [1978] : Manipulator Control Using The Configuration Space Method. *Industrial Robot*, 5, pp. 69-73.
- Rößler, J. [1980] : A Decentralized Hierarchical Control Concept For Large-Scale Systems. *Proc. IFAC Symp. Large Scale Systems, Theory, & Applications, Toulouse, France*, Tittli, A., and Singh, M.G. (eds), pp. 171-179.
- Shahinpoor, M., [1987] : *A Robot Engineering Textbook*. Harper and Row Publ., New York.
- Shoureshi, R., Corless, M. J., and Roesler, M. D., [1987] : Control Of Industrial Manipulators With Bounded Uncertainties. *ASME J. Dynam. Syst. Meas. and Control*, 109, pp. 53-59.
- Shoureshi, R., Momot, M. E., and Roesler, M. D., [1990] : Robust Control For Manipulators With Uncertain Dynamics. *Automatica*, 26, pp. 353-359.

- Singh, M. G., [1977] : *Dynamical Hierarchical Control*. North Holland, Amsterdam.
- Slotine, J., and Sastry, S., [1983] : Tracking Control Of Nonlinear Systems Using Sliding Surfaces, With Application To robot Manipulators. *Int. J. Control*, 38, pp. 465-492.
- Song, Y. D., and Gao, W. B., [1991] : Path Tracking Control Of Robot Manipulators Via The VSS Approach. *Int. J. Systems Sci.*, 22, pp. 151-163.
- Spong, M. W., and Vidyasagar, M., [1989] : *Robot Dynamics And Control*. John Wiley and Sons, Inc., New York, New York.
- Stokic, D. M., and Vukobratovic, M. K., [1984] : Practical Stabilization Of Robotic Systems By Decentralized Control. *Automatica*, 20, pp. 353-358.
- Tarn, T. J., Bejczy, A. K., Yun, X., and Li, Z., [1988] : Nonlinear Feedback And Its Simulation For Robot Arms With Third Order Dynamic System Model. Robotics Lab Report, SSM-RL-88-09, Dept. of System Science and Math., Washington Univ., St. Louis, MO 63130, USA.
- Tarn, T. J., Bejczy, A. K., Yun, X., and Li, Z., [1991] : Effect Of Motor Dynamics On Nonlinear Feedback Robot Arm Control. *IEEE Trans. Robotics and Automation*, RA-7, pp. 114-122.
- Troch, I., [1986] : There Is A Need For Simulation Based On Implicit Models. *Proc. of the Second European Simulation Congress*, Antwerp, Belgium, pp. 157-162.
- Troch, I., Kopacek, P., Desoyer, K., and Breiteneker, F. [1986] : Modelling Robots For Controller Design. *Proc. JSST Conf. on Recent Advances in Simulation of Complex Systems*, Tokyo, Japan, pp. 164-169.

- Tourasis, V. D., and Newman, C., [1985] : Properties And Structure Of Dynamic Robot Models For Control Engineering Applications. *J. Mechanism and Machine Theory*, 20, pp. 27-40.
- Tzafestas, S., and Stavrakakis, G., [1986] : Model Reference Adaptive Control Of Industrial Robots With Actuator Dynamics. *Theory of Robots : Selected Papers from IFAC/IFIP/IMACS Symp.*, Vienna, Austria, Kopacek, P., Troch, I., and Desoyer, K. (eds), pp. 233-240.
- Vukobratovic, M. K., Hristic, D., and Stokic, D. M., [1980] : Dynamic Control Synthesis Of Industrial Manipulation Systems. *Proc. of the Tenth Int. Symp. on Indust. Robots, Fifth Conf. on Indust. Robot Tech.*, Milan, Italy, pp. 331-347.
- Vukobratovic, M. K., and Potkonjak, V., [1982] : *Scientific Fundamentals Of Robotics 1 : Dynamics Of Manipulation Robots - Theory And Applications*. Springer-Verlag, Berlin.
- Vukobratovic, M. K., and Stokic, D. M., [1982] : *Scientific Fundamentals Of Robotics 2 : Control Of Manipulation Robots - Theory And Applications*. Springer-Verlag, Berlin.
- Vukobratovic, M. K., and Stokic, D. M., [1983] : Suboptimal Synthesis Of A Robust Decentralized Control For Large-Scale Mechanical Systems. *Large-Scale Systems : Theory and Applications, Proc. of the IFAC/IFORS Symp.*, Warsaw, Poland, Straszak, A. (ed), Pergamon Press, New York, pp. 175-181.
- Vukobratovic, M. K., Stokic, D. M., and Kircanski, N., [1985] : *Scientific Fundamentals Of Robotics 5 : Non-Adaptive And Adaptive Control Of Manipulation Robots*. Springer-Verlag, Berlin.
- Whitney, D. E., [1972] : The Mathematics Of Coordinated Control Of Prosthetic Arms And Manipulators. *ASME J Dynam. Syst. Meas. and Control*, pp. 303-309.

- Wijesoma, S. W., and Richards, R. J., [1990] : Robust Trajectory Following Of Robots Using Computed Torque Structure With VSS. *Int. J. Control*, 52, pp. 935- 962.
- Wu, C. H. and Paul, R. P., [1982] : Resolved Motion Force Control Of Robot Manipulators. *IEEE Trans. Syst. Man and Cybernatics*, SMC-12, pp. 266-275.
- Young, K. K., [1978] : Controller Design For A Manipulator Using Theory Of Variable Structure Systems. *IEEE Trans. Syst. Meas. and Cybernetic*, SMC-8, pp. 101-109.
- Zainol Anuar, B. M. S., [1985] : *Modelling And Control Of Industrial Robots*. Ph.D. Thesis, City University, London, UK.
- Zaprjanov, J., and Boeva, S., [1981] : Hierarchical Decentralized Control Of Industrial Robots. *IFAC Control Science and Technology : Proc. of the Eighth Triennial World Congress*, Kyoto, Japan, Akashi, H. (ed), pp. 1979-1984.