

# ROBUST AND FAULT TOLERANT CONTROL OF MODULAR AND RECONFIGURABLE ROBOTS

by

Sajan Abdul

Master of Science in Engineering,  
Lakehead University,  
Thunder Bay, Ontario, Canada, 2004

A thesis

presented to Ryerson University  
in partial fulfillment of the  
requirements for the degree of

Doctor of Philosophy

in the Program of Aerospace Engineering

Toronto, Ontario, Canada, 2008

© Sajan Abdul 2008



Library and Archives  
Canada

Published Heritage  
Branch

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

Bibliothèque et  
Archives Canada

Direction du  
Patrimoine de l'édition

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file Votre référence*  
ISBN: 978-0-494-58992-2  
*Our file Notre référence*  
ISBN: 978-0-494-58992-2

#### NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

#### AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

  
**Canada**



## **Author's Declaration**

I hereby declare that I am the sole author of this thesis.

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

Sajan Abdul

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

Sajan Abdul



# ROBUST AND FAULT TOLERANT CONTROL OF MODULAR AND RECONFIGURABLE ROBOTS

Doctor of Philosophy  
In the Program of  
Aerospace Engineering  
2008

Sajan Abdul

School of Graduate Studies  
Ryerson University

## ABSTRACT

Modular and reconfigurable robot has been one of the main areas of robotics research in recent years due to its wide range of applications, especially in aerospace sector. Dynamic control of manipulators can be performed using joint torque sensing with little information of the link dynamics. From the modular robot perspective, this advantage offered by the torque sensor can be taken to enhance the modularity of the control system. Known modular robots though boast novel and diverse mechanical design on joint modules in one way or another, they still require the whole robot dynamic model for motion control, and modularity offered in the mechanical side does not offer any advantage in the control design.

In this work, a modular distributed control technique is formulated for modular and reconfigurable robots that can instantly adapt to robot reconfigurations. Under this control methodology, a modular and reconfigurable robot is stabilized joint by joint, and modules can be added or removed without the need of re-tuning the controller. Model uncertainties associated with load and links are compensated by the use of joint torque sensors. Other model uncertainties at each joint module are compensated by a decomposition based robust controller for each module. The proposed distributed control

technique offers a ‘modular’ approach, featuring a unique joint-by-joint control synthesis of the joint modules.

Fault tolerance and fault detection are formulated as a decentralized control problem for modular and reconfigurable robots in this thesis work. The modularity of the system is exploited to derive a strategy dependent only on a single joint module, while eliminating the need for the motion states of other joint modules. While the traditional fault tolerant and detection schemes are suitable for robots with the whole dynamic model, this proposed technique is ideal for modular and reconfigurable robots because of its modular nature. The proposed methods have been investigated with simulations and experimentally tested using a 3-DOF modular and reconfigurable robot.

## ACKNOWLEDGMENTS

Let me begin by expressing my gratitude to my research supervisor Prof. Guangjun Liu for having guided the work that led to this thesis. His constructive criticisms and suggestions have pushed me through these years of endeavor.

Also I would like to thank my Ph.D committee members, Prof. Vincent Chan, Prof. Krishna Dev Kumar, Prof. Puren Ouyang and Prof. Jeff Xi for their advice and comments. I wish to express my warm and sincere thanks to my thesis examiner, Prof. Ou Ma for his valuable remarks. Their suggestions have helped in improving this thesis to a great extent.

Thanks are offered to Natural Sciences and Engineering Research Council of Canada (NSERC) and Engineering Services Inc. (ESI) for financially supporting this project. I also want to thank the members of the Systems and Control Laboratory, Mathew Adamson, Xiaojia He, Jing Yuan and Daqing Wang for their help and support during these years. Among my colleagues at Ryerson, I would like to thank Godard, Kam Shahid, Surjit Varma for those relaxing coffee breaks, those discussions and conversations about virtually everything under the sky. I would like thank Liao Liang, Richard Mohammed and Jude Alexander for their companionship and good cheer.

The company of Kathik Narayan, Sudeep Sukumar, and Arun Ahluwalia was always enjoyable; with their witty comments and their ability to see the funnier side of any situation was always a source of fun to all our circle of friends. With all of them around there never was a dull moment and it was always stress relieving to be in their company. Our cricket team Demerara Southerners and our weekend league matches brought fun to my otherwise routine world of computers and controls.

Special thanks to Godard for bringing me in touch with some of the nicest and fun-loving people. Thanks to him, I will always cherish the time I spent in Toronto. These people and many more made my stay at Ryerson a memorable experience.

My deepest thanks to Mahesh, Girish and fellow thampurans for their calming presence and unwavering friendship. They were always a source of inspiration and have



contributed to my thesis by their very soothing presence and phone calls. Naming each one of them and listing everything I owe them would be a thesis in its own right.

I want to thank my family for being an unending source of support and love. I would also like to express my continuing debt of gratitude to my father, in this the sixth year of his passing. My mother who always encouraged us to study further was a big source of inspiration. Last, but definitely not the least, thanks to my wife and companion, Shinu, for her love and support.

*To my family*



## TABLE OF CONTENTS

Author's Declaration .....	iii
ABSTRACT .....	v
ACKNOWLEDGMENTS .....	vii
TABLE OF CONTENTS .....	xi
List of Tables .....	xv
List of Figures.....	xvii
List of Appendices.....	xxi
List of Nomenclature .....	xxiii
1. Introduction .....	1
1.1 Modular and Reconfigurable Robots .....	2
1.2 Fault Tolerant Control of MRR .....	4
1.3 Motivation and Objectives of Research .....	6
1.4 Contributions.....	7
1.5 Outline of Thesis .....	8
2. Dynamic Modeling.....	9
2.1 Model of Modular and Reconfigurable Robot.....	9
2.2 Representation of Faults.....	11
2.2.1 Actuator Fault .....	12
2.2.2 Sensor Fault.....	13
2.2.3 Process Fault .....	13
2.3 Redundancy in Fault Control Systems.....	14
2.3.1 Structural Redundancy based Fault Tolerance.....	14
2.3.2 Kinematic Redundancy .....	15
2.3.3 Functional Redundancy.....	15

3. Torque Sensor based Control.....	17
3.1 Distributed Control of Modular and Reconfigurable Robots.....	19
3.2 Simulation Results.....	28
4. Fault Tolerant Control of Modular and Reconfigurable Robot.....	33
4.1 Fault Tolerant Control Design.....	35
4.2 Fault Detection .....	40
4.2.1 Fault Detection Scheme .....	41
4.3 Simulation Results.....	44
4.3.1 Fault Tolerant Control.....	45
4.3.2 Fault Detection .....	47
5. Experiments .....	51
5.1 Experimental Setup .....	51
5.1.1 Control System Architecture.....	53
5.1.2 Hardware .....	53
5.1.3 Electrical System Architecture.....	54
5.1.4 Mechanical Setup .....	58
5.1.5 Software .....	58
5.2 Results for Torque Sensor based Control.....	60
5.3 Results for Decentralized Fault Tolerant Control .....	69
5.4 Results for Fault Detection Scheme .....	75
5.5 Discussions.....	77
6. Conclusions and Future Work .....	79
6.1 Conclusions .....	79
6.2 Future Work .....	80
Bibliography.....	83
Appendix A .....	93

Appendix B .....	95
B.1 Dynamic Model of 3-DOF planar manipulator.....	95
Appendix C .....	97
C.1 Controller Architecture and Flow Diagrams.....	97



## **List of Tables**

Table 3-1. Parameters of the simulated system .....	27
Table 4-1. Parameters of the simulated system .....	44
Table 5-1. Parameters of the system.....	60





## List of Figures

Figure 2-1. Schematic diagram of a joint module .....	9
Figure 2-2. A typical feedback control system.....	11
Figure 2-3. Actuator Fault .....	12
Figure 2-4. Sensor Fault .....	13
Figure 3-1. Position tracking errors of three joints.....	29
Figure 3-2. Position error of the third joint with torque sensor ripples .....	29
Figure 3-3. Position tracking error of the first joint with torque sensor nonlinearity.....	30
Figure 3-4. Position error on first joint with torque sensor delay.....	31
Figure 4-1. Fault detection and fault tolerance framework .....	34
Figure 4-2. The model-based analytical redundancy approach .....	40
Figure 4-3. Tracking errors under actuator faults(at $t = 7$ s on first joint and $t = 15$ s on third joint) .....	45
Figure 4-4. Changes in parameter $\theta_i^l$ under fault occurrence.....	46
Figure 4-5. Commanded torque at first joint .....	47
Figure 4-6. Velocity estimation error on the first joint with actuator fault at $t = 7$ s.....	48
Figure 4-7. Velocity estimation error on third joint, with actuator fault at $t = 15$ s .....	48
Figure 5-1. The components of harmonic drive transmission .....	52
Figure 5-2. Control system architecture for modular robot system.....	53
Figure 5-3. Electrical communication .....	56
Figure 5-4. Unfiltered ADC signal .....	57
Figure 5-5. Filtered ADC signal .....	57
Figure 5-6. A modular and reconfigurable robot system – Configuration 1 .....	59
Figure 5-7. Modular and reconfigurable robot - Configuration 2 .....	59
Figure 5-8. Trajectories for the three joints .....	61
Figure 5-9. Position tracking error at three joints with MRR in Configuration 1 (No payload) .....	62

Figure 5-10. Position tracking error at three joints (MRR in Configuration 1, payload added at $t=7s$ ).....	63
Figure 5-11. Position tracking error at three joints with MRR in Configuration 1 (payload at $t = 0 s$ ) .....	64
Figure 5-12. Commanded torque at three joints of MRR system (Configuration 1).....	65
Figure 5-13. Position tracking error at three joints, with no payload (MRR in Configuration 2).....	66
Figure 5-14. Position tracking error at three joints with payload at $t = 0 s$ (MRR in Configuration 2).....	67
Figure 5-15. Position tracking error at three joints with payload added at $t = 7 s$ (MRR in Configuration 2).....	68
Figure 5-16. Joint position error at three joints (fault free operation, MRR in Configuration 1).....	70
Figure 5-17. Joint position error at three joints (fault free operation, MRR in Configuration 2).....	71
Figure 5-18. Joint position error at three joints (fault occurrence at $t = 4 s$ and $t = 6 s$ , MRR in Configuration 1).....	72
Figure 5-19. Joint position error at three joints (fault occurrence at $t = 4 s$ and $t = 6 s$ , MRR in Configuration 2).....	73
Figure 5-20. Commanded torque for three joints (MRR in Configuration 1) .....	74
Figure 5-21. Changes in parameter $\theta_i^1$ under fault occurrence.....	75
Figure 5-22. Velocity estimation error with actuator fault at first joint (fault occurrence at $t = 4 s$ ) .....	76
Figure 5-23. Velocity estimation error with actuator fault at third joint (fault occurrence at $t = 6 s$ ) .....	77
Figure C-1. Control system architecture of modular and reconfigurable robot.....	97
Figure C-2. Host PC - DSP controller communication over CAN.....	97
Figure C-3. Architecture of MRR control system .....	98
Figure C-4. Flow chart for the initialization of MRR module (Level 1) .....	99

Figure C-5. Flow chart of the MRR module operation (Level 2).....	100
Figure C-6. Message interpreter and execution at DSP from communication layer .....	101
Figure C-7. Flow chart of sequence of tasks on DSP controller, at each sample period	102
Figure C-8. Flow chart for the operation of 3DOF MRR system.....	103



## **List of Appendices**

Appendix A.....	93
Appendix B.....	95
B.1    Dynamic Model of 3-DOF planar manipulator.....	95
Appendix C.....	97
C.1    Controller Architecture and Flow Diagrams.....	97



## List of Nomenclature

### Roman Symbols

$I_m$	Motor inertia
$F_q$	Position dependent friction
$q$	Joint position
$\dot{q}$	Joint velocity
$\ddot{q}$	Joint acceleration
$q_d$	Desired joint position
$\dot{q}_d$	Desired joint velocity
$\ddot{q}_d$	Desired joint acceleration
$q_r$	Mixed tracking error
$\tilde{q}$	Joint position error
$K_D$	Controller gain
$K_t$	Actuator effectiveness factor
$U_f$	Controller compensating effects of lower modules
$V$	Lyapunov function candidate
$\delta$	Coupling terms due lower modules
$B$	Viscous friction Coefficient
$F_s$	Static friction coefficient
$F_c$	Coulomb friction coefficient
$F_\tau$	Stribeck friction coefficient
$F$	Parametric uncertainty
$e$	Velocity estimation error
$z_m$	Unit vector along axis of rotation of motor
$z$	Unit vector along axis of rotation of joint
$K$	Observer gains



$u_u$	Compensator for effects of lower joint modules
$u_{pc}$	Compensator for the constant part of parametric uncertainty
$u_{pv}$	Compensator for the variable part of parametric uncertainty
$M$	Product of motor inertia and gear ratio
$v$	Estimated velocity signal

## Greek Symbols

$\tau_J$	Coupling torque
$\tau_s$	Sensed torque
$\gamma$	Gear ratio
$\Gamma$	Adaptive gain
$\tau$	Joint torque
$\tau_c$	Commanded torque
$\theta$	Parameter vector
$\tilde{\theta}$	Error in parameter vector
$\hat{\theta}$	Estimated parameter vector
$\varepsilon_{if}$	Threshold value on estimation error
$\lambda$	Positive gain
$\kappa$	Gain of the torque sensor
$\eta$	Gain of the torque sensor
$\mu$	Offset of the torque sensor

## Subscripts

$i$	Joint number
-----	--------------

## Superscripts

$T$	Transpose
$i$	Joint number





# Chapter 1

## Introduction

Many robotic manipulators have been developed for different practical applications, but most of them are not designed to provide the capability of reconfiguration and are difficult to adapt to various tasks and environments. This motivated the design of a reconfigurable robot system (Yim, 1994). The term modular and reconfigurable robot (MRR) is referred to a robot manipulator assembled from discrete mechanical joints and links into one of many possible manipulator structures. Modular and reconfigurable robots are characterised by interchangeable links and joint modules of various sizes. Using standard mechanical and electrical interfaces, the recombination of modules is carried out to create various robot configurations that meet a wide range of different task requirements.

Such a modular and reconfigurable robot system has several advantages over conventional manipulators:

- *Cost effectiveness*: lowers manufacturing cost, and ease of replacement brings about a reduction in cost.
- *Modularity*: introduces flexibility to robots by making them reconfigurable.
- *Manufacturability*: reduces the number of operations for an individual part and thus simplifies manufacturing; making them easier and cheaper to build.
- *Redundancy*: implies highly redundant systems since many modules are available due to the ease of manufacture, thus enhances diagnosability.
- *Repairability*: if a module fails, it is easy to replace the module since there are others which can take up the same job.
- *Durability*: against system malfunctions due to replaceable standardized units.

Some of the mechanical designs of MRR discussed in the literature, based on the prototype systems built in various research institutes, can be found in (Fukuda and Nakagawa, 1988, Matsumaru, 1995, Castano and Will, 2000, Xi, *et al.*, 2006).

“Can a modular control technique be designed for any modular and reconfigurable robot?” is the main question posed. Is the modularity achieved in the mechanical design achievable in the control design too? Can the control law be made independent of each new configuration of the robot? Can faults at each joint module be detected and tolerated with minimum or no information from the other modules? To answer these, two different cases are considered.

This thesis is presented in two sections, each of which addresses a different control problem and the ensuing control strategy. One section presents the distributed control strategy for a modular and reconfigurable robot. The other section presents a decentralized fault tolerant and fault detection scheme for the modular robot. Both the schemes are based on joint torque sensing and hence allow a joint-by-joint control strategy, *i.e.*, the control of one joint module does not require the motion states of the other modules and is independent of the other modules. This improves the modularity and repairability of the robots and makes the control procedure ideal for modular and reconfigurable robots.

## **1.1 Modular and Reconfigurable Robots**

With substantial application potential, especially in the aerospace sector, the development of modular and reconfigurable robot (MRR) is one of the most promising research areas in robotics (Yim, *et al.*, 2007). Three types of MRRs have been reported in the literature: self-assembly, self-configuring and manual-configuring. Self-assembly robots possess the highest level of reconfigurability because they can detach from and attach into a robotic system automatically (Fukuda and Nakagawa, 1988, Chirikjian, 1994, Tomita, *et al.*, 1996, Yim, *et al.*, 2000). Self-configuring robots cannot perform self-assembly, but they can perform reconfiguration after a robotic system is assembled with some form of manual assistance. Manual-configuring robots are in fact modular robots that can be reconfigured with some form of manual assistance (Hui, *et al.*, 1993).

A survey of MRR systems is discussed in the publication compiled by Setchi and Lagos (2004).

So far, reconfigurable systems are developed based on an ad hoc approach due to lack of general-purpose simulation and control techniques. Though such tools are not available, some groundwork has been done and archived in the literature. The main concept of developing reconfigurable robots is based on the use of modular components as building blocks. For this reason, various modules have been proposed for reconfigurable robots. However, these proposed modules are the traditional mechanical components, *i.e.*, joints and links. While the reported reconfigurable robots may represent excellent mechanical design concepts, the modules of known MRRs are not “modular” from control systems point of view because of the existence of dynamic coupling among the modules, which is left to be dealt with by the controller. Conventional robot control methods are based on a known robot configuration and its associated dynamic model, with limited allowable model uncertainties such as unknown payloads (Liu and Goldenberg, 1996a, 1996b, 1997). Even though in theory robust control schemes can handle large model uncertainties caused by robot reconfigurations, they are not practical because of the large uncertainty that leads to extremely high feedback gains that cannot be implemented due to limited structure rigidity, computer sampling rate, actuator bandwidth and saturation, etc. Innovative design and control methods are in need in order to develop truly modular and reconfigurable robots. Melek and Goldenberg (2003) recently proposed a neurofuzzy control approach for MRRs, which uses learning control to compensate unmodelled system dynamics due to reconfiguration. The controller parameters are updated using a skill module that is a part of the higher level of the control system hierarchy. While the effectiveness of this approach has been experimentally demonstrated, a difficulty may limit its practical application, which is associated with the initial stage right after a reconfiguration and before the learning controller has learnt the unmodelled system dynamics. At this stage, the behaviour of the robot is not predictable.

Recent development in MRR control includes a position control scheme of MRR discussed by Melek and Najjaran (2005), with consideration of external disturbance.

Varying payload at the end-effector is treated as external disturbance, and a mathematical formulation connecting end-effector tracking error and payload is derived. The dynamic control of MRR using a virtual decomposition based control approach is discussed by Zhu and Lamarche (2007).

## 1.2 Fault Tolerant Control of MRR

Dynamic control of manipulators can be performed using joint torque sensing (Hashimoto, 1989, Kosuge *et al.*, 1990, Imura, *et al.*, 1991), without the need for modeling of link dynamics. The effectiveness of these approaches depends on accurate joint-torque sensing. Since joint torque sensor gains and offsets are susceptible to changes due to varying temperature and other factors and onboard calibration of joint torque sensors is difficult, it is desirable to estimate torque sensor parameters. In (Aghili and Namvar, 2006) the dynamical equation is parameterized such that torque sensor parameters are included in the overall system parameters to be estimated. This approach in other words would accept uncalibrated torque sensor signals for the controller. In (Liu, *et al.*, 2008), a distributed control technique for modular and reconfigurable robots is developed based on joint torque sensing, enabling the joint by joint stabilization of the modular robot and allowing instant adaptation to robot reconfigurations. As there is no coupling effect left on the base joint after the feedback of torque sensor signal, this joint can be stabilized using any control design technique for a single joint, such as decomposition based control scheme (Liu and Goldenberg, 1996a, 1996b). Once the base joint is stabilized independently, the acceleration and velocity of this joint must be bounded and can only cause bounded uncertainty to upper joints. The bounded uncertainty is then compensated at upper modules to achieve stabilization of the succeeding joint. Proceeding similarly, the upper modules are stabilized.

Based on the above mentioned approach, this strategy is aimed to achieve fault detection and fault tolerance at individual MRR modules, so that potential faults are dealt with at the module level and a faulty module can be repaired or replaced independent of the rest modules. Most approaches of fault tolerant control in robot manipulators are centered on the addition of some form of redundancy like those based on actuators (Wu, *et al.*, 1993, Sreevijayan, *et al.*, 1994), joints (Ting, *et al.*, 1993, Shin

and Lee, 1999), sensors (Visinsky, *et al.*, 1995), to name a few. An alternate way of achieving redundancy is by means of analytical relationships among system variables. This form of redundancy termed as analytical redundancy has received significant attention in the past (Gertler, 1998, Frank, 1990, Isermann, 1984, Frank, *et al.*, 2000). Several such approaches to fault tolerance of robots have been proposed, such as observer-based approaches (Caccavale and Walker, 1997, Filaretov, *et al.*, 1999, Vemuri and Polycarpou, 1997, Hammouri, *et al.*, 1999, De Persis and Isidori, 2001), parity based (Staroswiecki and Comtet-Varga, 2001) and parameter estimation based methods (Dixon, *et al.*, 2000, Isermann, 1993, Zhang, *et al.*, 1998). In (Caccavale and Walker, 1997) residuals are generated by comparing the predicted observer outputs with the measured system outputs. However, in these observer-based schemes, the measurement or estimation of acceleration signals are necessary. In (Dixon, *et al.*, 2000) a filtered torque estimation based technique is used to eliminate the requirement of acceleration measurement. All these techniques are designed for robot manipulators with fixed configuration, and are not based on distributed control schemes. In (Liu, 2001) an adaptive robot control strategy with consideration of actuator faults is proposed, which incorporates actuator effectiveness factors in a model parameterization with commanded torque as the input. A more recent adaptive fault detection scheme is presented in (De Luca and Mattone, 2004) based on the use of generalized momenta (De Luca and Mattone, 2003). In (Zhang, *et al.*, 2002) a fault tolerant architecture for nonlinear uncertain dynamic systems is proposed based on non-linear adaptive estimators. This approach depends on adaptive parameter estimates for fault detection.

Some recent results based on adaptive and robust observers can be found in (Jiang, *et al.*, 2002, 2004). Methods using learning approaches are presented in (Vemuri, 2001, Zhang, *et al.*, 2002). A fault detection scheme with online estimators is investigated in (Vemuri, 2001) to estimate the unknown constant sensor bias for diagnosis of sensor faults. Research on fault detection and fault identification of robot manipulators using discrete-time observers are reported in (Caccavale and Walker, 1997, Caccavale, 1998), non-linear adaptive state estimation in (Dixon, *et al.*, 2000), nonlinear observers in (McIntyre, *et al.*, 2005) and parity relation based approach in (Filaretov, *et al.*, 2003) .



However, the above approaches all require the link dynamics for the detection algorithm and are based on the whole robot dynamics.

### 1.3 Motivation and Objectives of Research

Various modules of MRRs have been developed recently in research institutions and industry worldwide. However, the developed modules are traditional in terms of electro-mechanical components, *i.e.*, joints and links. While the reported reconfigurable robots may represent excellent electro-mechanical design concepts, the modules of known MRRs are not “modular” in terms of software and control because of the existence of dynamic coupling among the modules. This phenomenon is left to be dealt with by the control system, and very limited work has been reported in this domain. Conventional robot control methods are designed for robots with fixed configurations, based on known kinematics and dynamics models, and they are not suitable to handle the large model uncertainties associated with reconfigurations of MRRs. To a large extent, the realization of the application potential of MRRs is limited by the lagging in the development of synergetic control systems.

The key motivating factors for this research are as noted:

- In (Imura, *et al.*, 1991), it is pointed out that the use of torque sensor can reduce the robot dynamic equations to purely a mathematical problem of motor control with link dynamics taken care using the torque sensor readings. This motivates the joint torque sensing based control of MRRs, with the uncertainties caused by reconfiguration.
- Many publications can be found on fault tolerance of robot manipulators and others on possibilities of MRRs being used for better fault tolerant capabilities. However, there is very minimal reference on the actual applications of fault tolerance control on MRRs. Therefore, one of the objectives of this research will be to apply the fault tolerant techniques to a practical MRR system.

In this thesis, MRR system control techniques to realize the application potential of MRRs are developed. The objectives are:

- to develop a modular control architecture based on joint torque sensing and decoupled modular control techniques for MRRs to achieve automatic control system adaptation to MRR reconfigurations, and
- to synthesize modular actuator fault detection and fault tolerant control methods.

## 1.4 Contributions

The following contributions were made from this research:

- Stabilization and control of modular robots using distributed control law  
A distributive approach with a unique joint-by-joint control technique of joint modules was designed. This approach uses a joint torque sensor to compensate for the uncertainties associated with load and link, and hence a control law independent of link dynamics.
- Torque sensor based fault tolerant control of modular robots.  
A fault tolerant control was devised based on joint torque sensor that offers a strategy dependent on single joint module. The fault tolerance at each joint module will not require motion states of other joint modules and made independent of other joint module controllers.
- Decentralized fault detection strategy for modular robots.  
Since the use of joint torque sensors offers modularity for system, this is used in the design of fault detection technique. The detection scheme uses a velocity estimation error to detect the occurrence of faults.
- Modular fault tolerant control law under actuator degradation.  
The devised actuator fault tolerant control is devised as decentralized control technique, such that actuator degradation at each joint module is

compensated at module level. The control law at each module will not require the motion states of other joint modules.

## **1.5 Outline of Thesis**

In the present study, a tracking control for modular and reconfigurable robots are developed. To investigate this, two different cases are considered. In the first case a  $n$ -DOF modular robot, with torque sensor at each joint module, is considered. A distributed control law that can achieve overall stability of the robot by a joint-by-joint control approach is devised. The second case involves a modular fault tolerant and fault detection technique, ideally suited to improve the repairability of a modular robot system. Here fault at each joints are detected and corrective actions taken at module level.

In Chapter 2 the dynamic model of the modular robot is discussed. The control design and formulation of the distributed control strategy for the  $n$ -DOF modular and reconfigurable robot is presented in Chapter 3. The chapter also includes the simulation results for a 3-DOF robot stabilization case. Chapter 4 discusses the fault tolerant control and a design of decentralized fault tolerant and fault detection of modular and reconfigurable robots. The simulation results for the fault tolerant and fault detection schemes are also discussed in this chapter. In Chapter 5, the experimental setup and experimental results for the control schemes are presented. The conclusions are given in Chapter 6.

## Chapter 2

### Dynamic Modeling

#### 2.1 Model of Modular and Reconfigurable Robot

In this section a dynamic model of a robot, employing a joint torque feedback, is derived. Most existing robot arms with  $N$  degrees of freedom are modeled as serial link mechanisms consisting of  $N$  rigid links and  $N$  single axis joints. Each link of such an arm is considered to be driven directly or indirectly by an actuator.

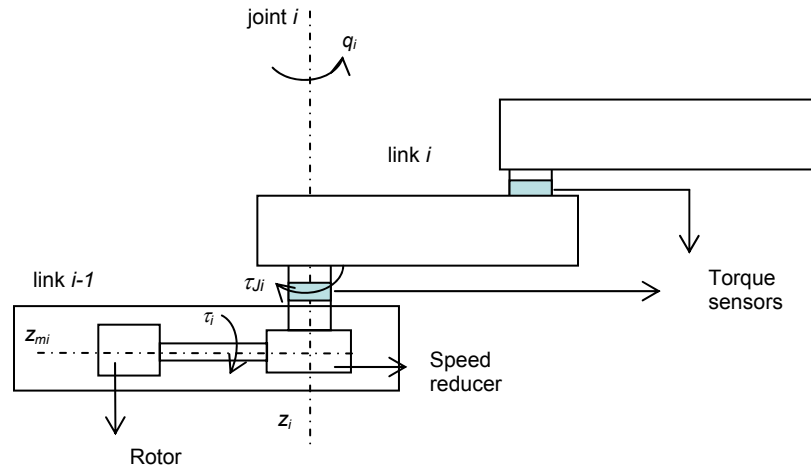


Figure 2-1. Schematic diagram of a joint module

We consider modular and reconfigurable robots constructed with  $n$  modules, and each module is integrated with a rotary joint with a speed reducer and a torque sensor as illustrated in Figure 2-1. The dynamics of each module is divided into two, namely the motor system and the link system, with the joint torque sensor as the separator. The motor system includes a rotor and a speed reducer and the link system is composed of

the link. The  $i^{th}$  motor which drives the  $i^{th}$  joint is fixed to the  $(i-1)^{th}$  link. The coupling force at the joint is measured using joint torque sensor on the  $i^{th}$  joint.

For each module, we assume

- A1. The rotor is symmetric with respect to the axis of rotation.
- A2. The joint flexibility is negligible.
- A3. The torque transmission does not fail at the speed reducer, and the inertia between the torque sensor and the speed reducer is negligible.

For a modular and reconfigurable articulated robot with modules installed in series, each module provides a rotary joint. The base module is denoted the first module. Modules close to the first module are named lower modules, and modules close to the end-effector are called upper modules.

For the  $i^{th}$  module, we use the following notations:

$I_{mi}$  : rotor moment of inertia about the axis of rotation;

$\gamma_i$  : reduction ratio of the speed reducer ( $\gamma_i \geq 1$ );

$q_i$  : joint angle;

$f_i(q_i, \dot{q}_i)$  : joint friction;

$\tau_{ji}$  : coupling torque at the torque sensor location;

$\tau_i$  : output torque of the rotor;

$z_{mi}$  : unit vector along the axis of rotation of the  $i^{th}$  rotor;

$z_i$  : unit vector along the axis of rotation of joint  $i$ .

Based on the dynamic equations of rigid robot manipulators with  $n$  rotary joints and joint torque sensing derived in (Imura, *et al.*, 1991), we formulate the dynamic equation of each module as follows (given in Appendix A):

For the base module,  $i = 1$ ,

$$I_{m1}\gamma_1\ddot{q}_1 + f_1(q_1, \dot{q}_1) + (\tau_{J1}/\gamma_1) = \tau_1 \quad (2.1)$$

For the second module from the base,  $i = 2$ ,

$$I_{m2}\gamma_2\ddot{q}_2 + f_2(q_i, \dot{q}_i) + I_{m2}z_{m2}^T z_1\ddot{q}_1 + (\tau_{J2}/\gamma_2) = \tau_2 \quad (2.2)$$

For  $i \geq 3$

$$\begin{aligned} & I_{mi}\gamma_i\ddot{q}_i + f_i(q_i, \dot{q}_i) + I_{mi} \sum_{j=1}^{i-1} z_{mi}^T z_j \ddot{q}_j \\ & + I_{mi} \sum_{j=2}^{i-1} \sum_{k=1}^{j-1} z_{mi}^T (z_k \times z_j) \dot{q}_k \dot{q}_j + (\tau_{Ji}/\gamma_i) = \tau_i \end{aligned} \quad (2.3)$$

The joint friction,  $f_i(q_i, \dot{q}_i)$ , is modeled as in (Liu, 2002, Liu, *et al.*, 2004)

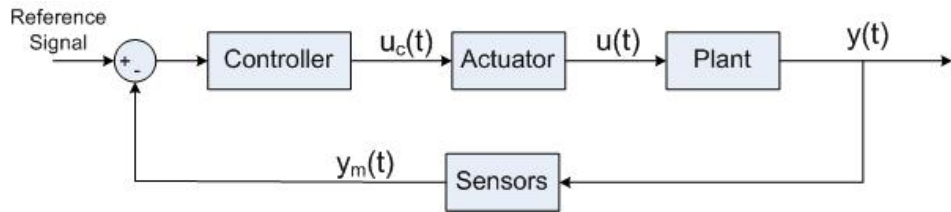
$$f_i(q_i, \dot{q}_i) = B_i \dot{q}_i + F_q(q_i, \dot{q}_i) + \left( F_{ci} + F_{si} \exp(-F_{\tau i} \dot{q}_i^2) \right) \text{sat}(\dot{q}_i) \quad (2.4)$$

where  $F_{ci}$  denotes the Coulomb friction related parameter,  $B_i$  denotes the viscous friction coefficient,  $F_{si}$  denotes the static friction related parameter,  $F_{\tau i}$  is a positive parameter related to the Stribeck effect. The saturation function is defined as

$$\text{sat}(\dot{q}_i) = \begin{cases} 1 & \text{for } \dot{q}_i > 0 \\ 0 & \text{for } \dot{q}_i = 0 \\ -1 & \text{for } \dot{q}_i < 0 \end{cases} \quad (2.5)$$

and  $F_q(q_i, \dot{q}_i)$  denotes the position dependent friction and other friction modeling errors.

## 2.2 Representation of Faults



**Figure 2-2. A typical feedback control system**

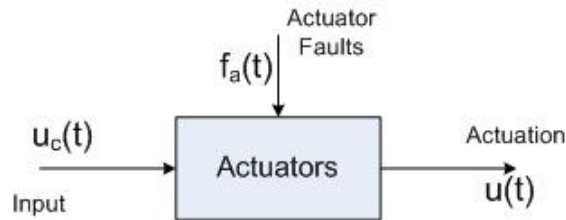
A typical control system, as shown in Figure 2-2, consists of sensors, actuators, controllers and the physical plant being controlled. Any abnormal operations among these components are referred to as faults. Faults may be categorized based on their physical locations in control systems as either sensor faults, actuator faults, plant faults or controller faults or based on system characteristics as additive faults or multiplicative faults. Fault classifications based on physical locations have been adopted in this thesis.

### 2.2.1 Actuator Fault

Actuator faults are usually characterized as discrepancies between the expected actuator output and its actual output, or complete loss of control capabilities. This may result from aging or worn out components in the actuator, an element being stuck at some specific value and/or a change in dynamic characteristics. The actuator input is usually passed through a digital to analog converter in computer controlled systems. Thus, digital to analog converters can also be considered as a part of the actuator block.

The actuator output  $u(t)$  of the system is often not directly accessible, and is the actuator response to an actuator command  $u_c(t)$  from the controller. When an actuator fault occurs, any change in the actuator input will not result in the desired response at the actuator output. Typically an actuator fault can be represented as shown in Figure 2-3. The actuator fault vector is represented by  $f_a(t)$ .

$$u(t) = u_c(t) - f_a(t) \quad (2.6)$$



**Figure 2-3. Actuator Fault**

Alternatively the fault at the actuators can be described by an actuator effectiveness factor (Liu, 2001) which would represent the transmission rate of the commanded actuator signal to the actuator response.

$$u(t) = K_t u_c(t) \quad (2.7)$$

where  $K_t$  is the actuator effectiveness factor. And  $K_t = 1$  indicate the fault free condition and  $K_t = 0$  indicates the complete failure of the actuator *i.e.*, no torque gets transmitted to the load side.

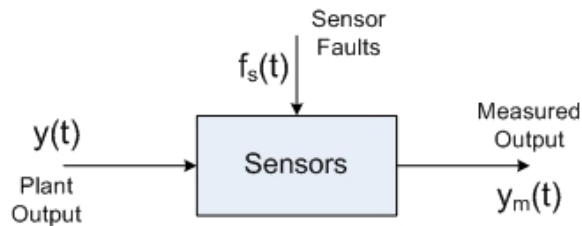
### 2.2.2 Sensor Fault

The sensor faults are incorrect readings due to malfunctions in sensory elements or transducers, unknown bias at the sensor output or unexpected changes in the dynamic characteristics of the transducers. Since the signal from a sensor carries the most important information in feedback control systems, the state of sensor determines the reliable operation of the system. In a digital computer controlled system, the sensor signals will usually pass through analog to digital converters before used by the control system. Hence, the analog to digital converters can be considered to be a part of the sensor block.

The actual outputs of the system  $y(t)$  are measured through sensors. The sensor faults are shown in Figure 2-4 and can be described mathematically as an additive term

$$y(t) = y_m(t) - f_s(t) \quad (2.8)$$

where  $f_s(t)$  is the sensor fault vector.



**Figure 2-4. Sensor Fault**

### 2.2.3 Process Fault

The plant or system faults are caused by changes in the plant parameters or the plant dynamics. System performances are affected by these changes.



## **2.3 Redundancy in Fault Control Systems**

Redundancy refers to the additional system resources, such as actuators, sensors or controllers which are more than what are required to achieve the control objective during a normal operation. The existing literature on the research in robot fault tolerance is concentrated in the following fields:

### **2.3.1 Structural Redundancy based Fault Tolerance**

#### **2.3.1.1 Actuator Redundancy**

In a practical system actuators are used to manipulate control signals to operate a process. The type and power levels of the actuators generally depend on the specific applications. For motion control applications, the actuators used typically includes stepper motors, DC servo motors, linear motors, pneumatic or hydraulic motors. The mechanical motions of the actuator can result in wearing and aging of the actuator components. This can result in degradation of the actuator and deterioration of actuator effectiveness.

Most of the early works on fault tolerance of mechanical failures in robots has concentrated on algorithms which relied on duplicated parts for their fault tolerant capabilities. In (Tesar, 1990; Wu, *et al.*, 1993) investigations into the methods of duplicating motors in a joint have been discussed. The two motors in a joint work together to provide single output velocity to the joint, and when one of the motor fails, the other takes over the functions of the faulty motor. The fault tolerant advantages of redundancy have also led to adding extra structures like backup arm (Tesar, 1990; Ting, *et al.*, 1993), allowing reconfiguration possibilities in the event of failures. But in robotics, the amount of physical redundancy is often limited by cost, size and power considerations. This limitation in structural redundancy is complemented to a certain extent by the kinematic and analytical redundancy techniques. The structural redundancy can be broadly classified into actuator and sensor redundancies.

### **2.3.1.2 Sensor Redundancy**

Sensors are physical devices that measure various physical variables, such as position, pressure, torque, temperature etc., with the associated circuits. Generally the signal level at a sensor output is relatively low and hence addition signal conditioning and amplification are necessary. Faults in the sensor or its associated circuitry are often classified as sensor faults. The physical quantities act as the input to the sensor and the sensor outputs are conditioned to provide the outputs.

One of the widely used redundant sensor scheme is the triple modular redundancy (TMR) which consists of three identical sensors employed to measure the same physical quantity. The outputs of the three sensors are compared using a voting circuit. If one sensor output differs significantly from the other two, then it is considered to be a faulty signal. But the use of identical sensors can pose a disadvantage that an identical fault can result because of manufacturing defect or design error. To avoid this scenario, dissimilar sensors are often used with additional circuitry to process the signals before they are sent to the voting circuit.

### **2.3.2 Kinematic Redundancy**

Many of the present day robots have the advantage of being kinematically redundant, *i.e.*, the robot has more degrees of motion or freedom than is required to position its end effector along working trajectories. This property is usually exploited in designing fault tolerant algorithms which use alternate configurations to recover the robot in the presence of joint failures. These algorithms (Maciejewski, 1990) would use the existing structure rather than the use of extra motor, sensors or other hardware components. In (Ting, *et al.*, 1994; Chen, *et al.*, 1998; Park, *et al.*, 2003) a torque redistribution technique is discussed while joint velocity redistribution technique is used in (Chen, *et al.*, 1999).

### **2.3.3 Functional Redundancy**

If an accurate mathematical model of the physical system is available, the output of system can be computed, and the difference between the system output and computed system output should be small. This difference termed residual signal, will show a

change in value with the occurrence of fault. In cases where the robots have no redundant hardware or kinematic redundancy, functional redundancy techniques are used which will exploit the existing structure for fault tolerance using functionally equivalent data from dissimilar components (Visinsky, *et al.*, 1993). An observer based fault diagnosis technique is discussed in (Filaretov, *et al.*, 1999), which uses nonlinear observers for residual generation.

## Chapter 3

### Torque Sensor based Control

Model based robot control schemes have been proposed in the past for achieving tracking control, like resolved acceleration control (Luh, *et al.*, 1980) or the computed torque control techniques (Craig, 1989, Slotine and Li, 1991, Spong and Vidyasagar, 1989). These approaches depend on the precise knowledge of the robot parameters and of the possibly varying load; they may perform poorly when the model is not accurate. Adaptive control of manipulators was proposed (Middleton and Goodwin, 1988) to deal with parametric uncertainties. However, the large number of parameters associated with the link dynamics and over-parameterization makes it difficult to meet the persistency excitation condition necessary for parametric convergence and achieving high performance (Ioannou and Sun, 1996). In addition, model-based approaches tend to increase the computational burden (Craig, 1989).

Alternatively, the dynamic control of manipulators can be performed using joint torque feedback without the need for modeling link dynamics. This reduces the computational burden by dramatically reducing the number of parameters for adaptive schemes. The benefits of using torque sensory feedback to improve the performance of robotic systems have been recognized in the robotics community (Stokic and Vukobratovic, 1993). One of the earlier works related to torque sensor based position control of robots is discussed in (Kosuge, 1987, Hashimoto, 1989). (Vischer and Khatib, 1995) discusses the design of high performance torque controller joints, based on contactless inductive transducers as compared to strain gauge sensors. Design of harmonic drive gears with built-in torque sensing is dealt in (Godler and Hashimoto, 1998, Hashimoto, *et al.*, 1993). Joint torque feedback can be used to compensate the nonlinearities and modeling uncertainties of manipulator dynamics (Hashimoto 1989) or simply those of actuators (Zhang and Furusho, 1998). (Hashimoto, *et al.*, 1993) used the

elasticity of the flexspline in a harmonic drive to measure the joint torque. This technique has the advantage of using the existing structural flexibility of the robots.

Joint torque feedback can, in theory be used to eliminate completely the effect of external torque disturbances and load torques. This requires the system be equipped with built-in torque sensing to measure the load torque (Hashimoto, 1989, Kosuge, *et al.*, 1990).

(Kosuge, *et al.*, 1990) demonstrated experimentally the effectiveness of joint torque feedback to compensate the entire link dynamics of a SCARA type direct-drive robot. The actuator dynamics was ignored and a simple control law with good robustness against varying loads was proposed. Hashimoto (1989) applied this technique to an actuator geared with a harmonic drive. The dynamic coupling terms in the robot dynamics are claimed to be small at higher gear ratios and hence treated as small disturbances. A survey of joint torque feedback can be found in (Stokic and Vukobratovic, 1993).

In general, joint torque feedback cannot achieve dynamic decoupling between joints unless the manipulator has joints with high gear ratios (Hashimoto, 1989). However, the effect of link dynamics is especially at lower gear ratios (Stokic and Vukobratovic, 1993) and this is where the joint torque feedback is particularly useful. Furthermore, the effectiveness of the joint torque feedback in suppressing the effect of linkage torque dynamics on the motion critically relies on accurate joint-torque sensing. Different techniques have been used for joint torque measurement such equipping manipulators joints with joint torque sensor device (Aghili, *et al.*, 2001) or mounting strain gauges on flexspline of harmonic drives (Hashimoto, *et al.*, 1993).

In this thesis, a control system architecture is developed for modular and reconfigurable articulated robots that can instantly adapt to robot reconfigurations and can control the reconfigured robot without having to adjust controller parameters. A joint torque sensor is embedded within each module, and the torque sensor measurement is utilized to automatically compensate for coupling effect. This concept is built upon published research results on robust control of conventional robot manipulators using joint torque sensors (Kosuge, *et al.*, 1990, Imura, *et al.*, 1991). However, the known

torque sensor based approaches use centralized control techniques and are inadequate for control of modular robots, which calls for distributed control for each stand alone module. The proposed method stabilizes an MRR joint by joint, rather than controlling the robot as a whole (Liu, *et al.*, 2006, 2008).

### 3.1 Distributed Control of Modular and Reconfigurable Robots

The dynamic equation (2.3) can be rewritten as

$$I_{mi}\gamma_i\ddot{q}_i + f_i(q_i, \dot{q}_i) + \delta_i + (\tau_{Ji}/\gamma_i) = \tau_i \quad (3.1)$$

The term  $\delta_i$  is constituted of the coupling effects of lower joints on the  $i^{th}$  joint, given by

$$\delta_i = I_{mi} \sum_{j=1}^{i-1} z_{mi}^T z_j \ddot{q}_j + I_{mi} \sum_{j=2}^{i-1} \sum_{k=1}^{j-1} z_{mi}^T (z_k \times z_j) \dot{q}_k \dot{q}_j \quad (3.2)$$

This term is a source of model uncertainty, which depends on the accelerations and velocities of all the lower  $i-1$  joints, and it needs to be compensated using a robust controller at the  $i^{th}$  joint.

The following properties are used in the subsequent analysis and design of the control law:

*Property 1:* The accelerations and velocities of the stabilized joints are bounded. Hence the following upper bounds exist:

$$|\delta_i| \leq \rho_{Di} \quad (3.3)$$

*Property 2:* Since the term  $F_q(q_i, \dot{q}_i)$  is bounded the following upper bounds exist

$$|F_q(q_i, \dot{q}_i)| \leq \rho_f \quad (3.4)$$

From (3.3) - (3.4), the following upper bounds can be defined:

$$|\delta_i + F_q(q_i, \dot{q}_i)| \leq \rho_{Fi}$$

Define the overall control of each joint as

$$\tau_i = \frac{\tau_{J_i}}{\gamma_i} + u_i \quad i = 1, 2, \dots, n \quad (3.5)$$

where  $u_i$  is a new control input to be determined for the  $i^{th}$  joint.

For the first (base) joint,  $i = 1$ , combining (2.1) with (3.5) yields

$$I_{m1}\gamma_1\ddot{q}_1 + f_1(q_1, \dot{q}_1) = u_1$$

As there does not exist any coupling effect on the base joint, the control input  $\tau_1$  can be obtained using control design techniques for a single joint, such as the decomposition-based robust control scheme (Liu, 2002).

For the second joint,  $i = 2$ , combining (2.2) with (3.5) yields

$$I_{m2}\gamma_2\ddot{q}_2 + f_2(q_2, \dot{q}_2) + I_{m2}z_{m2}^T z_1 \ddot{q}_1 = u_2$$

The inertial force associated with acceleration of the first joint is involved in the above equation, and model uncertainty exists in  $z_{m2}^T z_1$  as a result of reconfiguration. However, as the first joint has been stabilized independently, the acceleration of the first joint must be bounded. Hence, the uncertainty in the term  $I_{m2}z_{m2}^T z_1 \ddot{q}_1$  is also bounded, and such bounded model uncertainty can be compensated with a robust control scheme.

For  $i \geq 3$ , substituting (2.3) into (3.5) yields

$$\begin{aligned} I_{mi}\gamma_i\ddot{q}_i + f_i(q_i, \dot{q}_i) + I_{mi} \sum_{j=1}^{i-1} z_{mi}^T z_j \ddot{q}_j \quad i = 3, 4, \dots, n \\ + I_{mi} \sum_{j=2}^{i-1} \sum_{k=1}^{j-1} z_{mi}^T (z_k \times z_j) \dot{q}_k \dot{q}_j = u_i \end{aligned} \quad (3.6)$$

Substituting, (3.2) in (3.6) yields,

$$I_{mi}\gamma_i\ddot{q}_i + f_i(q_i, \dot{q}_i) + \delta_i = u_i$$

Examining (3.6), one can find that the motion of upper joints does not affect the lower joints, which has been compensated using the torque measurement in (3.5). However, the motion of the lower joints still affects upper joints through inertial and Coriolis coupling forces. This can be readily understood by analyzing the dynamic equation of the third joint,  $i = 3$ ,

$$\begin{aligned}
& I_{m3}\gamma_3\ddot{q}_3 + I_{m3}\left[z_{m3}^T z_1 \ddot{q}_1 + z_{m3}^T z_2 \ddot{q}_2\right] \\
& + I_{m3}z_{m3}^T(z_1 \times z_2)\dot{q}_1\dot{q}_2 + f_3(q_3, \dot{q}_3) = u_3
\end{aligned} \tag{3.7}$$

Inertial and Coriolis forces associated with the motion of the first and second joints are involved in (3.7), and reconfiguration can result in model uncertainties in the terms  $z_{m3}^T z_1$ ,  $z_{m3}^T z_2$  and  $z_{m3}^T(z_1 \times z_2)$ . However, as the first and second joints have been stabilized, the accelerations and velocities of the first and second joints must be bounded. Hence, the uncertainties in  $I_{m3}\left[z_{m3}^T z_1 \ddot{q}_1 + z_{m3}^T z_2 \ddot{q}_2\right]$  and  $I_{m3}z_{m3}^T(z_1 \times z_2)\dot{q}_1\dot{q}_2$  are also bounded and could be compensated with robust control schemes. Expanding the observation to the general case (3.6), for the  $i^{th}$  joint, the model uncertainties in the terms  $I_{mi}\sum_{j=1}^{i-1} z_{mi}^T z_j \ddot{q}_j$  and  $I_{mi}\sum_{j=2}^{i-1}\sum_{k=1}^{j-1} z_{mi}^T(z_k \times z_j)\dot{q}_k\dot{q}_j$  are bounded and could be compensated with robust control schemes.

Based on the above observation, it can be concluded that the control input  $u_i$  can be designed joint by joint, with the consideration of the bounded model uncertainty due to the motion of the lower joints. This conclusion forms the basis of the proposed control design in this paper.

In theory, a saturation-based robust controller can compensate for bounded model uncertainty. However, high feedback gain is required in order to achieve high accuracy, which is always limited by hardware issues including unmodeled high order plant dynamics and sensor measurement noise. The key in practical robust control design is to achieve desired performance with minimum feedback gains. To this end, a decomposition based control design approach is developed in (Liu and Goldenberg, 1996a, 1996b). The fundamental strategy of the decomposition based system modeling and control approach is to distinguish between uncertain parameters and variables of different physical types, and to design a separate compensator for each of them, while taking into account each specific physical feature. This approach advocates treating each type of model uncertainty with the most suitable and efficient means, including PID, robust, adaptive, and sensor based control methods. Robust compensators are used only



to compensate for uncertainties that cannot be estimated or measured in real time. The overall controller is generated by a synergetic integration of these compensators.

The decomposition based control approach is applied to design a robust control scheme for the system under consideration. Examining the equation (3.6), if we assume the joint velocities and accelerations are available, the model uncertainties in the above two terms are all due to the unknown misalignment between the joint and motor axes. However, while it is reasonable to assume the joint velocities are measurable, it is not practical to assume the measurement of joint accelerations, which is avoided in the proposed control method.

The following are defined first for the control design

$$\begin{aligned}\tilde{q}_i &= q_i - q_{di}, \\ q_{ri} &= \dot{\tilde{q}}_i + \lambda_i \tilde{q}_i, \\ q_{ai} &= \ddot{q}_{di} - 2\lambda_i \dot{\tilde{q}}_i - \lambda_i^2 \tilde{q}_i\end{aligned}\tag{3.8}$$

where  $\lambda_i$  is a positive constant.

It is assumed that some of the system parameters are unknown but the inertia of motor  $I_{mi}$  is known. It is also assumed that the reference trajectory, its first and second derivatives are bounded.

The friction of the  $i^{th}$  joint is compensated using the scheme developed in (Liu, 2002), where  $Y(\dot{q}_i)$  and  ${}^i\tilde{F}$  are defined as

$$\begin{aligned}Y(\dot{q}_i) &= \begin{bmatrix} \dot{q}_i & \text{sgn}(\dot{q}_i) & \text{sgn}(\dot{q}_i) \exp(-\hat{F}_{\tau i} \dot{q}_i) & -\hat{F}_{si} \dot{q}_i^2 \text{sgn}(\dot{q}_i) \exp(-\hat{F}_{\tau i} \dot{q}_i) \end{bmatrix} \\ {}^i\tilde{F} &= \begin{bmatrix} \hat{B}_i - B_i & \hat{F}_{ci} - F_{ci} & \hat{F}_{si} - F_{si} & \hat{F}_{\tau i} - F_{\tau i} \end{bmatrix}^T\end{aligned}\tag{3.9}$$

If the parametric uncertainty  ${}^i\tilde{F}$  is considered unknown but constant, this uncertainty can be compensated using an integrator type compensator. However, in practice, the parametric model uncertainty may not always be constant, due to temperature and lubrication changes.

To incorporate variable parametric model uncertainty compensation,  ${}^i\tilde{F}$  is decomposed as

$${}^i\tilde{F} = {}^i\tilde{F}_c + {}^i\tilde{F}_v \quad (3.10)$$

where  ${}^i\tilde{F}_c$  is a constant unknown vector, and  ${}^i\tilde{F}_v$  is variable and bounded as follows:

$$\left| {}^i\tilde{F}_{vn} \right| < {}^i\rho_n \quad n = 1, 2, 3, 4. \quad (3.11)$$

Applying the approach of decomposition based control design in (Liu, 2002) an adaptive compensator is designed to compensate the constant parametric uncertainty  ${}^i\tilde{F}_c$ , and a robust compensator for  ${}^i\tilde{F}_v$ .

For stabilizing the first joint, the following control torque is determined:

$$\begin{aligned} \tau_1 = & I_{m1}\gamma_1\dot{q}_{a1} + \hat{B}_1\dot{q}_1 + \left( \hat{F}_{c1} + \hat{F}_{s1} \exp(-\hat{F}_{\tau1}\dot{q}_1^2) \right) \text{sgn}(\dot{q}_1) \\ & + {}^1u_u + Y(\dot{q}_1)({}^1u_{pc} + {}^1u_{pv}) + \frac{\tau_{J1}}{\gamma_1} \end{aligned} \quad (3.12)$$

where  $I_{m1}$  is the motor inertia for the first joint,  $\tau_{J1}$  is the coupling torque at the torque sensor location,  $\hat{B}_1, \hat{F}_{c1}, \hat{F}_{s1}, \hat{F}_{\tau1}$  are the nominal friction parameters,  ${}^1u_u$  is the term designed to compensate for the non-parametric uncertainty  $F_{qi}(q, \dot{q})$  and the term  $\delta_i$  constituted by the lower modules. The terms  ${}^1u_{pc}$  and  ${}^1u_{pv}$  are designed to compensate for the parametric uncertainty  ${}^i\tilde{F}_c$  and  ${}^i\tilde{F}_v$ , respectively. The friction compensation is of the same form for all the joints, and hence for the  $i^{th}$  joint the compensators  ${}^iu_{pc}$ ,  ${}^iu_{pv}$  and  ${}^iu_u$  are defined by

$${}^iu_u = \begin{cases} -\rho_{Fi} \frac{q_{ri}}{|q_{ri}|} & |q_{ri}| > {}^i\varepsilon \\ -\rho_{Fi} \frac{q_{ri}}{{}^i\varepsilon} & |q_{ri}| \leq {}^i\varepsilon \end{cases} \quad (3.13)$$

$$\begin{aligned}
{}^i u_{pc} &= -k \int_0^t Y(\dot{q}_i)^T q_{ri} d\tau \\
{}^i u_{pvn} &= \begin{cases} -{}^i \rho_n \frac{{}^i \zeta_n}{|{}^i \zeta_n|} & |{}^i \zeta_n| > {}^i \varepsilon_{pn} \\ -{}^i \rho_n \frac{{}^i \zeta_n}{{}^i \varepsilon_{pn}} & |{}^i \zeta_n| \leq {}^i \varepsilon_{pn} \end{cases} \quad n=1,2,3,4.
\end{aligned} \tag{3.14}$$

where  ${}^i \zeta_n = Y(\dot{q}_i)^T q_{ri}$ , and  ${}^i \varepsilon$ ,  ${}^i \varepsilon_{pn}$  are positive control parameters.

The stabilization of the first joint, using control law given by (3.12), results in the uniformly ultimately boundedness of tracking error and thus the boundedness of the magnitudes of  $\dot{q}_1$  and  $\ddot{q}_1$ . Since  $\ddot{q}_1$  is bounded, a compensator designed using saturation based robust control could be used to compensate for the effects of  $\ddot{q}_1$ . Thus the control torque  $\tau_2$  for the second joint would be given by the control law discussed in (Liu, 2002) with an additional term  ${}^2 u_u$  to compensate for the effects of  $\ddot{q}_1$  and  $F_{qi}(q, \dot{q})$ .

$$\begin{aligned}
\tau_2 &= I_{m2} \gamma_2 q_{a2} + \hat{B} \dot{q}_2 + \left( \hat{F}_c + \hat{F}_s \exp(-\hat{F}_\tau \dot{q}_2^2) \right) \text{sgn}(\dot{q}_2) \\
&\quad + {}^2 u_u + Y(\dot{q}_2) ({}^2 u_{pc} + {}^2 u_{pv}) + \frac{\tau_{J2}}{\gamma_2}
\end{aligned} \tag{3.15}$$

Similarly for the  $i^{th}$  joint, we have control torque  $\tau_i$  given by

$$\begin{aligned}
\tau_i &= I_{mi} \gamma_i q_{ai} + \hat{B} \dot{q}_i + \left( \hat{F}_c + \hat{F}_s \exp(-\hat{F}_\tau \dot{q}_i^2) \right) \text{sgn}(\dot{q}_i) \\
&\quad + {}^i u_u + Y(\dot{q}_i) ({}^i u_{pc} + {}^i u_{pv}) + \frac{\tau_{Ji}}{\gamma_i}
\end{aligned} \tag{3.16}$$

Substituting (3.16) into (3.6) yields the closed loop expression for the  $i^{th}$  joint as

$$\begin{aligned}
M_i \dot{q}_{ri} + \lambda_i M_i q_{ri} &= Y(\dot{q}_i) (\tilde{F}_c + {}^i u_{pc}) + Y(\dot{q}_i) (\tilde{F}_v + {}^i u_{pv}) \\
&\quad + {}^i u_u - F_{qi}(q_i, \dot{q}_i) - \delta_i
\end{aligned} \tag{3.17}$$

where  $M_i = I_{mi} \gamma_i$ .

**Theorem 3.1:** *Given an  $n$ -DOF modular robot, with joint dynamics as given in (2.1) – (2.3) and the uncertainty defined by (3.2) and (3.4), the tracking error of each joint is uniformly ultimately bounded under the control law defined by (3.16). The ultimate bound of the tracking error is determined by the non-parametric uncertainty and control parameters only, and it is not affected by the parametric uncertainty.*

**Proof:**

A Lyapunov function candidate is defined as

$$V = \frac{1}{2} M_i q_{ri}^2 + \frac{1}{2} k \Psi_i^T \Psi_i \quad (3.18)$$

where  $\Psi_i = \frac{1}{k} \tilde{F}_c - \int_0^t Y(\dot{q}_i)^T q_{ri} d\tau$ ,

Since  $k$  and  $\tilde{F}_c$  are constants,  $\dot{\Psi}_i$  are given by  $\dot{\Psi}_i = -Y(\dot{q}_i)^T q_{ri}$ . Differentiating (3.18) yields

$$\begin{aligned} \dot{V} &= M_i q_{ri} \dot{q}_{ri} + k \Psi_i^T \dot{\Psi}_i \\ &= q_{ri} \left( -\lambda_i M_i q_{ri} + {}^i u_u - F_{qi}(q_i, \dot{q}_i) - \delta_i + \right. \\ &\quad \left. Y(\dot{q}_i)(\tilde{F}_c + {}^i u_{pc}) + Y(\dot{q}_i)(\tilde{F}_v + {}^i u_{pv}) \right) - k \Psi_i^T Y(q_i)^T q_{ri} \\ &= -\lambda_i M_i q_{ri}^2 + q_{ri} Y(\dot{q}_i)(\tilde{F}_v + {}^i u_{pv}) + q_{ri} Y(\dot{q}_i)(\tilde{F}_c + {}^i u_{pc}) \\ &\quad + q_{ri} ({}^i u_u - F_{qi}(q_i, \dot{q}_i) - \delta_i) - q_{ri} Y(\dot{q}_i) \left( \tilde{F}_c - k \int_0^t Y(\dot{q}_i)^T q_{ri} d\tau \right) \\ &= -\lambda_i M_i q_{ri}^2 + q_{ri} Y(\dot{q}_i)(\tilde{F}_v + {}^i u_{pv}) + q_{ri} ({}^i u_u - F_{qi}(q_i, \dot{q}_i) - \delta_i) \end{aligned}$$

If  $|q_{ri}| \geq {}^i \varepsilon$  then,  $q_{ri} ({}^i u_u - F_{qi}(q_i, \dot{q}_i) - \delta_i) = -|q_{ri}| \left( \rho_{Fi} - (F_{qi}(q_i, \dot{q}_i) + \delta_i) \frac{q_{ri}}{|q_{ri}|} \right)$

$$< 0$$

If  $|q_{ri}| < {}^i \varepsilon$  then,

$$\begin{aligned}
q_{ri} \left( {}^i u_u - F_{qi}(q_i, \dot{q}_i) - \delta_i \right) \\
= -q_{ri} \left( \rho_{Fi} \frac{q_{ri}}{{}^i \varepsilon} + F_{qi}(q_i, \dot{q}_i) + \delta_i \right) \\
= -\rho_{Fi} \frac{q_{ri}^2}{{}^i \varepsilon} - \left( F_{qi}(q_i, \dot{q}_i) + \delta_i \right) q_{ri} \\
\leq -\rho_{Fi} \frac{q_{ri}^2}{{}^i \varepsilon} + \rho_{Fi} |q_{ri}|
\end{aligned} \tag{3.19}$$

$$\begin{aligned}
q_{ri} Y(\dot{q}_i)(\tilde{F}_v + {}^i u_{pv}) &= \sum_{n=1}^4 q_{ri} Y_n(\tilde{F}_{vn} + {}^i u_{pvn}) \\
&= \sum_{n=1}^4 {}^i \zeta_n (\tilde{F}_{vn} + {}^i u_{pvn})
\end{aligned}$$

For  $i^{th}$  joint, and  $n = 1, 2, 3, 4$ , if  $|{}^i \zeta_n| > {}^i \varepsilon_{pn}$ ,

$${}^i \zeta_n (\tilde{F}_{vn} + {}^i u_{pvn}) = {}^i \zeta_n \left( \tilde{F}_{vn} - {}^i \rho_n \frac{{}^i \zeta_n}{|{}^i \zeta_n|} \right) < 0 \tag{3.20}$$

If  $|{}^i \zeta_n| \leq {}^i \varepsilon_{pn}$

$$\begin{aligned}
{}^i \zeta_n (F_{vn} + {}^i u_{pvn}) &= {}^i \zeta_n \left( \tilde{F}_{vn} - {}^i \rho_n \frac{{}^i \zeta_n}{|{}^i \zeta_n|} \right) \\
&\leq {}^i \zeta_n^i \left( {}^i \rho_n \frac{{}^i \zeta_n}{|{}^i \zeta_n|} - {}^i \rho_n \frac{{}^i \zeta_n}{\varepsilon_{pn}^i} \right)
\end{aligned} \tag{3.21}$$

Since the last term of (3.19) and (3.21) achieves a maximum value at  $|q_{ri}| \leq {}^i \varepsilon/2$  and

$|{}^i \zeta_{jn}| \leq {}^i \varepsilon_{pn}/2$  respectively, we have

$$\dot{V} \leq -\lambda_i M_i q_{ri}^2 + \frac{\rho_{Fi} {}^i \varepsilon}{4} + \sum_{n=1}^4 \frac{{}^i \rho_n {}^i \varepsilon_{pn}}{4} \tag{3.22}$$

From (3.22), it can be concluded that a Lyapunov function can be found if only

$$|r_i| > \sqrt{\frac{\left(\rho_{Fi}^i \varepsilon + \sum_{n=1}^4 \rho_n^i \varepsilon_{pn}\right)}{4\lambda_i M_i}}.$$

Define

$$S = \left\{ q_{ri} \in R^1 \mid q_{ri}^2 \leq \left( \rho_{Fi}^i \varepsilon + \sum_{n=1}^4 \rho_n^i \varepsilon_{pn} \right) / (2\lambda_i M_i) \right\}$$

Then on the surface of  $S, \partial S$ , we have

$$\dot{V} \leq - \left( \rho_{Fi}^i \varepsilon + \sum_{n=1}^4 \rho_n^i \varepsilon_{pn} \right) / 4.$$

Denote  $T$  as the time for the solution trajectory to intersect the surface  $\partial S$ , then

$$V(q_{ri}(T)) - V(q_{ri}(0)) \leq - \left( \left( \rho_{Fi}^i \varepsilon + \sum_{n=1}^4 \rho_n^i \varepsilon_{pn} \right) / 4 \right) T$$

Hence

$$T \leq \frac{4(V(q_{ri}(T)) - V(q_{ri}(0)))}{\left( \rho_{Fi}^i \varepsilon + \sum_{n=1}^4 \rho_n^i \varepsilon_{pn} \right)}$$

The boundedness of  $q_{ri}$  implies the boundedness of  $\tilde{q}_i$  and  $\dot{\tilde{q}}_i$  (Slotine and Li, 1991).

**Table 3-1. Parameters of the simulated system**

	Link 1	Link 2	Link 3
Mass of link (kg)	8	5	4
Length of link (m)	1	1	1
Link inertia (kg-m <sup>2</sup> )	1.0	0.8	0.6
Dist. to centre of mass (m)	0.5	0.5	0.5
Rotor inertia (kg-m <sup>2</sup> )	0.4	0.2	0.1
Gear reduction ratio	10	10	10

### 3.2 Simulation Results

To study the effectiveness of the proposed distributed control law, fault detection and fault tolerant control methods, a 3-DOF serial robot manipulator, working on a horizontal plane with the following parameters, is used for the simulations:

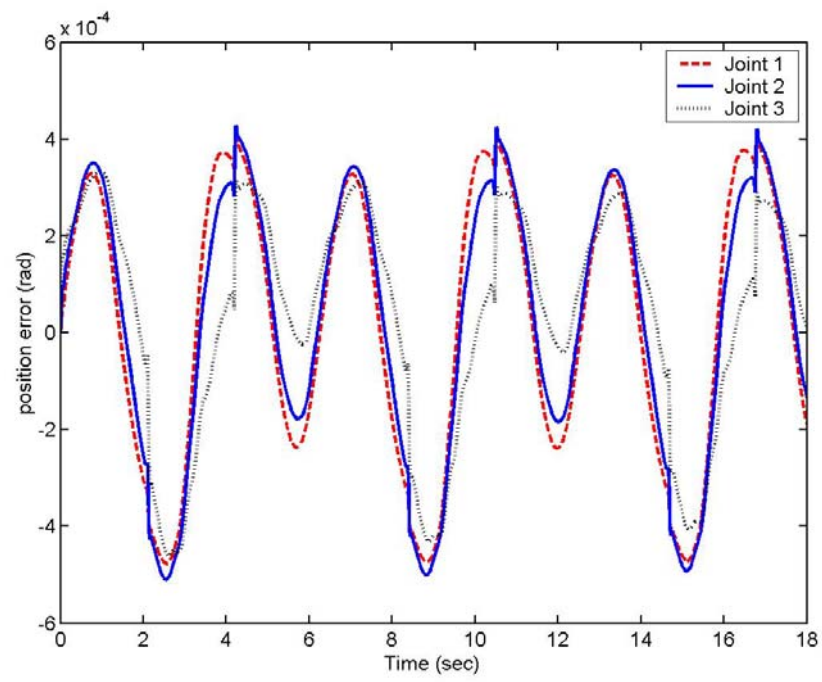
$$\begin{aligned} F_{\tau i} &= 100 \text{ s}^2 / \text{rad}^2, B_i = 1.5 \text{ N ms} / \text{rad}, F_{ci} = 3.5 \text{ Nm}, F_{si} = 1 \text{ Nm}, \\ K_{Di} &= 100, \kappa_i = 0.1, l_i = 0, \lambda_i = 100, \Gamma = 1 \text{ I}^{6 \times 6}, \varepsilon = 0.1, \rho = 2, \rho_F = 1 \end{aligned} \quad (3.23)$$

The dynamic equations of the model are as given in Appendix B. The nominal parameters of the friction model are assumed as  $\hat{B}_i = 1.2 \text{ Nms} / \text{rad}$ ,  $\hat{F}_{ci} = 1 \text{ Nm}$ ,  $\hat{F}_{\tau i} = 80 \text{ s}^2 / \text{rad}^2$ ,  $\hat{F}_{si} = 0.8 \text{ Nm}$ ,  $\hat{\kappa}_i = 0.2$ ,  $\hat{l}_i = 0.5$ . For simplicity, the same friction model and parameters were considered for all the three joints. For the simulations, the parameters of the manipulator are chosen as given in Table 3-1.

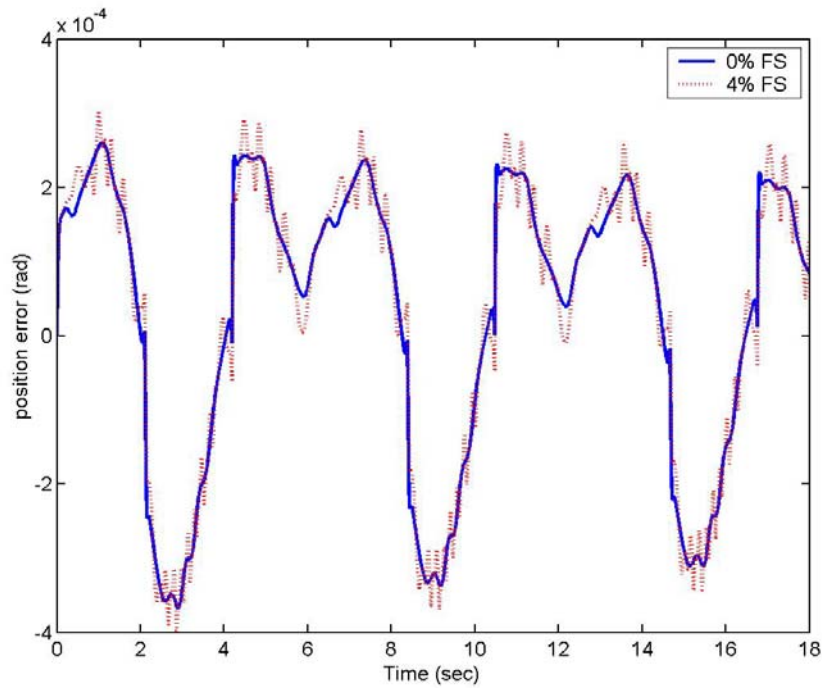
The desired trajectories for the three joints are selected as  $q_d = \sin(t) - 0.5 \sin(2t)$  for  $0 \leq t \leq 18 \text{ s}$ . The following controller parameters are used for the simulations

$$\begin{aligned} F_{\tau i} &= 100 \text{ s}^2 / \text{rad}^2, B_i = 1.5 \text{ N ms} / \text{rad}, F_{ci} = 3.5 \text{ Nm}, F_{si} = 1 \text{ Nm}, \\ K_i &= 1, \lambda_i = 80, {}^i \varepsilon = 0.1, \rho = 2, \\ {}^i \varepsilon_{p1} &= 0.01, {}^i \varepsilon_{p2} = 0.01, {}^i \varepsilon_{p3} = 0.01, {}^i \varepsilon_{p4} = 0.01, \\ {}^i \rho_1 &= 0.3 \text{ Nm s} / \text{rad}, {}^i \rho_2 = 1 \text{ Nm}, {}^i \rho_3 = 0.7 \text{ Nm}, {}^i \rho_4 = 20 \text{ s}^2 / \text{rad}^2, \\ \rho_{F1} &= 0.5, \rho_{F2} = 1.1, \rho_{F3} = 1.5 \end{aligned}$$

The plots in Figure 3-1 shows the tracking error for each joints under the control law defined in (3.16).

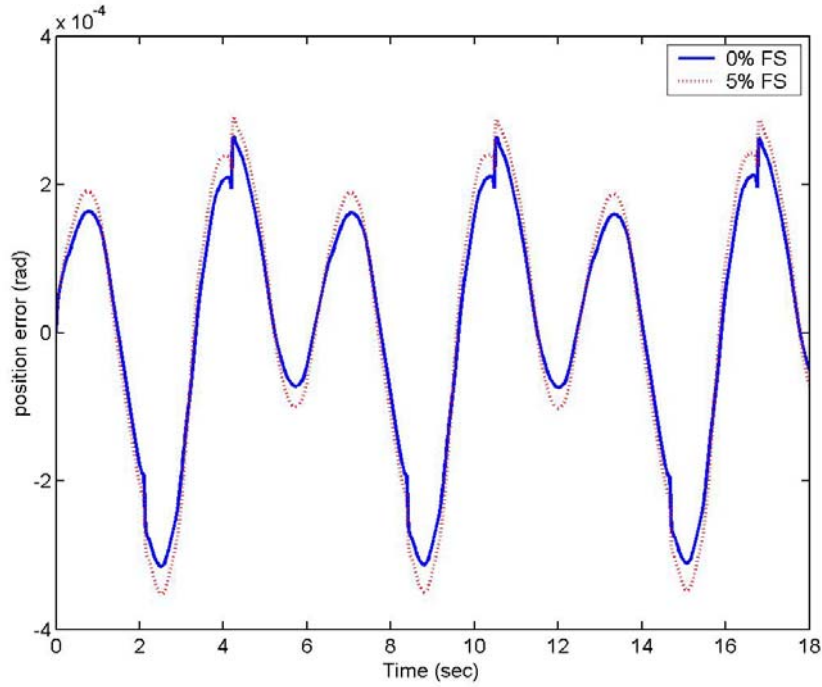


**Figure 3-1. Position tracking errors of three joints**



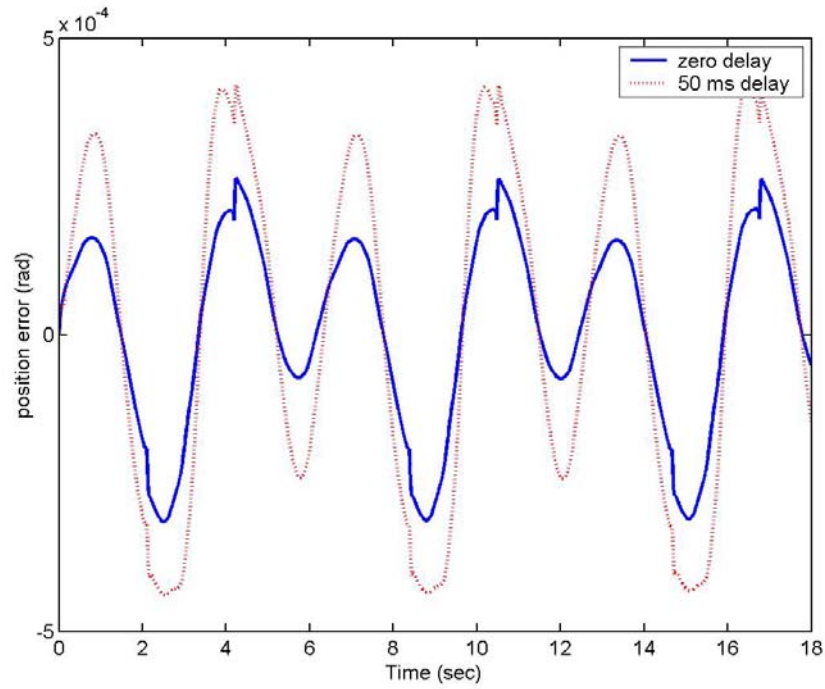
**Figure 3-2. Position error of the third joint with torque sensor ripples**





**Figure 3-3. Position tracking error of the first joint with torque sensor nonlinearity**

To study the tracking error variations sensor dynamics, simulations are carried out for different values of linearity and ripple factors of the torque sensor. The effect of ripple on tracking errors is studied, with the resulting error plot for third joint, as shown in Figure 3-2. The effect of nonlinearity on tracking error of the first joint is as shown in Figure 3-3. Since practical torque sensors have a delay in signal transmission, simulations were also carried with torque sensor delays. The tracking performance on first joint in the presence of delay in torque sensor is shown in Figure 3-4. The tracking errors for the other joints showed similar trends under the different values of ripple, nonlinearity and torque sensor delays. The controller was observed to be able to compensate for the errors introduced by the sensor dynamics. For larger time delays, it is seen that the tracking errors are higher due to the additional time lags introduced by the sensor delay. During simulations, it was observed that with higher gear ratio, the effects of torque sensor nonlinearities were minimal as expected.



**Figure 3-4. Position error on first joint with torque sensor delay**

The obtained results prove the effectiveness of the joint-by-joint stabilization scheme for control of modular and reconfigurable robot. The model uncertainties associated with link and payload masses are compensated using joint torque measurement, and the remaining uncertainties including the dynamic coupling effects and joint friction are compensated by the decomposition based robust controller. The simulation results also demonstrate the robustness of the controller against dynamic effects of the torque sensor.



## Chapter 4

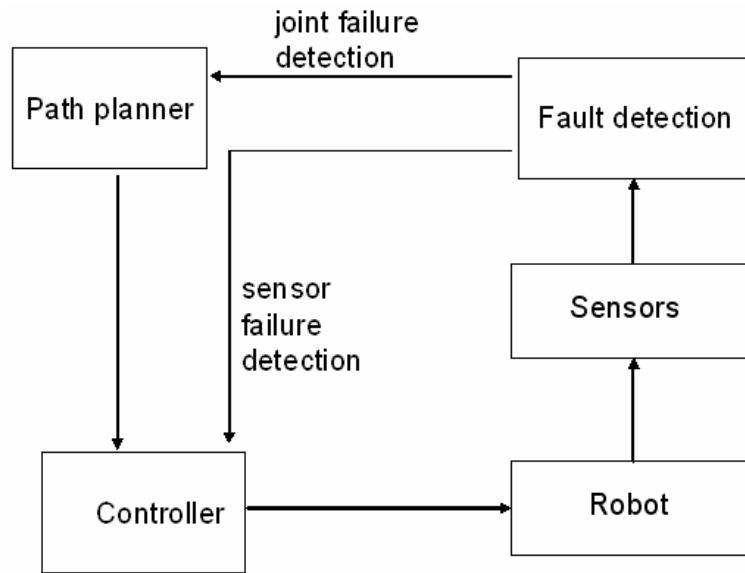
### Fault Tolerant Control of Modular and Reconfigurable Robot

Fault detection, isolation and fault tolerant control have recently been generating considerable interest in the control engineering community, in various fields of application. Aerospace and computer based systems are the two key areas where fault tolerance has shown its ever increasing importance. In robotics, its implementation in the field of space applications has attracted more research interests. A four level canonical architecture for fault tolerance is developed in (Chladek, 1990; Tesar, *et al.*, 1990) and these levels are listed as

- Dual joint actuators
- Parallel structured modules
- Redundant manipulators
- Multiple cooperating arms

In each of these levels, redundancy is incorporated to meet the demanding objectives of fault tolerance.

Fault tolerant robots are needed which can effectively detect and adapt to software and hardware failures to avoid catastrophic failures at the higher system level, like aerospace or nuclear applications. Though, in general the mechanical structure of robots comprising of links and joints typically has the capability to move around one axis, physical redundancy can offer some fault tolerance capability. But this increases the size of the robot and the associated manufacturing cost. Hence, instead of redundant motors for a single joint, the robots have redundancy in degrees of freedom which allows multiple joint configurations for every end effector position. This would allow the robot manipulator to withstand joint failures without the complete loss of working range. A typical framework for robot fault detection is as shown in Figure 4-1.



**Figure 4-1. Fault detection and fault tolerance framework**

For designing a fault tolerant control strategy for a robot manipulator, it is essential that detection or estimation of the fault signal is guaranteed. The fault signal might be due to an abrupt or incipient fault, or it might be a degradation or saturation of the actuators (Ting, *et al.*, 1993). The control strategies are devised based on this information of the faults. Dynamic fault tolerance and fault detection systems that can monitor the robot and compensate the faults in real-time can contribute significantly to system reliability and safety.

In this work, a decentralized fault tolerant control and fault detection schemes of modular robots are developed based on a joint-by-joint approach (Abdul and Liu, 2008a, 2007). In the proposed fault tolerant control of MRRs, actuator degradation at each joint module is tolerated independently of the other modules and fault detection schemes. For the proposed fault detection that is run in parallel with the fault tolerant control algorithm, a threshold based comparison on joint velocity estimation error is used to indicate the occurrence of a fault at each module. The threshold is determined based on the estimation error bounds obtained during fault free operation of the robot system. A fault is declared when the estimation error exceeds this threshold. Since the proposed scheme is a joint by joint scheme, rather than a scheme taking into account the whole robot dynamics, it is ideal for fault detection in modular robots. Faults are detected and

corrective measures can be taken at the module level (Abdul and Liu, 2008a, 2008b). The addition or removal of a module will not affect the control of other modules, enhancing the modularity and repairability of the overall MRR system. The paper presents a robust adaptive control scheme that can maintain the required performance in the presence of actuator degradation at joints and can detect faulty joint modules so that they can be replaced. This technique is different from most approaches which detect faults first and then use the information of detected faults in the operation of fault tolerant control. This can cause unknown transients due to the delay in the detection algorithm. The fault tolerant scheme proposed here is independent of the fault detection and does not rely on the fault detection information for the operation of fault tolerant control. This effectively avoids the chances of delay in fault tolerant control due to the delays caused by the detection algorithm. In the proposed fault tolerant control scheme, an adaptation law is used to friction parameters and the torque sensor related parameters. Each of the joint controllers does not require the motion states of other joints or link dynamics, and each joint is controlled independently from other joint modules, making it suitable for control of modular robots.

#### 4.1 Fault Tolerant Control Design

In the fault tolerant control design, we consider the actuation fault represented by change in the actuator effectiveness factor and assume that both the position and torque sensors are fault free and that delays in sensors are negligible.

The actual joint torques measured using torque sensors are given by

$$\tau_{Ji} = \eta_i \tau_{si} + \mu_i \quad (4.1)$$

where  $\eta_i$  and  $\mu_i$  are sensor gain and offset, respectively, and  $\tau_{si}$  denotes the torque sensor output signal. The dynamic equation (3.1) can then be rewritten as

$$I_{mi} \gamma_i \ddot{q}_i + \delta_i + f_i(q_i, \dot{q}_i) + \kappa_i \tau_{si} + l_i = K_{ti} \tau_{ci} \quad (4.2)$$

where  $\kappa_i = \eta_i / \gamma_i$ ,  $l_i = \mu_i / \gamma_i$ , and  $K_{ti}$ ,  $\tau_{ci}$  are the actuator effectiveness factor and commanded torque of the  $i^{th}$  motor. If  $K_{ti} = 1$ , the actuator is fault free, and  $K_{ti} = 0$  indicates complete failure of the actuator. The aim of the controller is to tolerate the faults before the actuator effectiveness factor reduces to zero, *i.e.*, before the complete failure of actuator.

The model (4.2) can be rewritten in a more compact form

$$Y_i(\ddot{q}_i, \dot{q}_i, q_i, \tau_{si})P_i + \delta_i + F_q(q_i, \dot{q}_i) = K_{ti}\tau_{ci} \quad (4.3)$$

where

$$Y_i(\ddot{q}_i, \dot{q}_i, q_i, \tau_{si}) = \begin{bmatrix} \ddot{q}_i & \dot{q}_i & \text{sat}(\dot{q}_i) & \exp(-F_{\tau i} \dot{q}_i^2) \text{sat}(\dot{q}_i) & \tau_{si} & 1 \end{bmatrix} \quad (4.4)$$

$$P_i = \begin{bmatrix} M_i & B_i & F_{ci} & F_{si} & \kappa_i & l_i \end{bmatrix}^T$$

where  $M_i = I_{mi} \gamma_i$ .

From (4.3) we have for the  $i^{th}$  module

$$\begin{aligned} \tau_{ci} &= K_{ti}^{-1} Y_i(\ddot{q}_i, \dot{q}_i, q_i, \tau_{si}) P_i + K_{ti}^{-1} (\delta_i + F_q(q_i, \dot{q}_i)) \\ &= Y_i(\ddot{q}_i, \dot{q}_i, q_i, \tau_{si}) \theta_i + K_{ti}^{-1} (\delta_i + F_q(q_i, \dot{q}_i)) \end{aligned} \quad (4.5)$$

with

$$\theta_i = \begin{bmatrix} K_{ti}^{-1} M_i & K_{ti}^{-1} B_i & K_{ti}^{-1} F_{ci} & K_{ti}^{-1} F_{si} & K_{ti}^{-1} \kappa_i & K_{ti}^{-1} l_i \end{bmatrix}^T.$$

Denote  $\hat{\theta}_i$  as the estimate of the parameter vector  $\theta_i$ . Then the uncertainties in joint parameters and actuator effectiveness factors are contained in the parameter estimation error  $\tilde{\theta}_i$ , which is given by

$$\tilde{\theta}_i = \hat{\theta}_i - \theta_i \quad (4.6)$$

The joint-by-joint stabilization of the modular robot has been achieved using the distributed control approach based on joint torque sensing (Liu, *et al.*, 2008).

From (3.3) – (3.4) and definition of  $K_{ti}^{-1}$ , the following upper bounds can be defined:

$$\left| K_{ti}^{-1} \delta_i \right| \leq \rho_{Dfi} \quad (4.7)$$

$$\left| K_{ti}^{-1} F_q(q_i, \dot{q}_i) \right| \leq \rho_{fq} \quad (4.8)$$

$$\left| K_{ti}^{-1} \delta_i + K_{ti}^{-1} F_q(q_i, \dot{q}_i) \right| \leq \rho \quad (4.9)$$

The following signals are defined

$$\begin{aligned} \tilde{q}_i &= q_i - q_{di}, & q_{vi} &= \dot{q}_{di} - \lambda_i \tilde{q}_i \\ q_{ri} &= \dot{\tilde{q}}_i + \lambda_i \tilde{q}_i, & q_{ai} &= \ddot{q}_{di} - 2\lambda_i \dot{\tilde{q}}_i - \lambda_i^2 \tilde{q}_i \end{aligned} \quad (4.10)$$

where  $\lambda_i$  is a positive constant.

The control law is defined as

$$\tau_{ci} = Y_i(q_{ai}, q_{vi}, q_i, \tau_{si}) \hat{\theta}_i - K_{Di} q_{ri} + U_f \quad (4.11)$$

and the adaptation law is defined as

$$\dot{\hat{\theta}}_i = -\Gamma_i^{-1} Y_i^T(q_{ai}, q_{vi}, q_i, \tau_{si}) q_{ri} \quad (4.12)$$

where

$$Y_i(q_{ai}, q_{vi}, q_i, \tau_{si}) = \begin{bmatrix} q_{ai} & q_{vi} & \text{sat}(\dot{q}_i) & \exp(-F_{\tau i} \dot{q}_i^2) \text{sat}(\dot{q}_i) & \tau_{si} & 1 \end{bmatrix}$$

and  $\Gamma_i$  is a positive definite matrix and  $K_{Di} > 0$ . The control term  $U_f$  is used to compensate for the term  $\delta_i$  constituted of coupling effects from the lower modules and friction term  $F_q(q_i, \dot{q}_i)$  (Liu and Goldenberg, 1996a, Liu, *et al.*, 2004).

$$U_f = \begin{cases} -\rho(q_{ri}/|q_{ri}|) & |q_{ri}| > \varepsilon \\ -\rho(q_{ri}/\varepsilon) & |q_{ri}| \leq \varepsilon \end{cases} \quad (4.13)$$

where  $\varepsilon$  is a positive control parameter.

From (4.11) and (4.6),

$$\begin{aligned} \tau_{ci} &= Y_i(q_{ai}, q_{vi}, q_i, \tau_{si}) (\tilde{\theta}_i + \theta_i) - K_{Di} q_{ri} + U_f \\ &= K_{ti}^{-1} \begin{pmatrix} M_i q_{ai} + B_i q_{vi} + F_{ci} \text{sat}(\dot{q}_i) + F_q(q_i, \dot{q}_i) + \\ F_{si} \exp(-F_{\tau i} \dot{q}_i^2) \text{sat}(\dot{q}_i) + \kappa_i \tau_{si} + l_i + \delta_i \end{pmatrix} \\ &\quad + Y_i(q_{ai}, q_{vi}, q_i, \tau_{si}) \tilde{\theta}_i - K_{Di} q_{ri} + U_f \end{aligned} \quad (4.14)$$

Combining (3.1), (4.5) and (4.14), we have the closed loop equation as

$$\begin{aligned} &K_{ti}^{-1} (M_i \dot{q}_{ri} + \lambda_i M_i q_{ri} + B_i q_{ri}) \\ &= Y_i(q_{ai}, q_{vi}, q_i, \tau_{si}) \tilde{\theta}_i - K_{Di} q_{ri} + U_f + K_{ti}^{-1} (F_q(q_i, \dot{q}_i) + \delta_i) \end{aligned} \quad (4.15)$$



**Theorem 4.1:** *Given an  $n$ -DOF modular robot, with joint dynamics as given in (2.1) – (2.3), actuator fault defined by variations of  $K_{ii}$  in (4.3) and the uncertainty defined in (4.6) and (4.9) the tracking error of each joint is uniformly ultimately bounded under the control law defined by (4.11).*

**Proof:**

A Lyapunov function candidate is defined as

$$V = \frac{1}{2} K_{ii}^{-1} M_i q_{ri}^2 + \frac{1}{2} \tilde{\theta}_i^T \Gamma_i \tilde{\theta}_i \quad (4.16)$$

Differentiating the above expression yields

$$\dot{V} = K_{ii}^{-1} M_i q_{ri} \dot{q}_{ri} + \tilde{\theta}_i^T \Gamma_i \dot{\tilde{\theta}}_i \quad (4.17)$$

Since the unknown parameters  $\theta_i$  is constant, we have

$$\dot{\tilde{\theta}}_i = \dot{\theta}_i \quad (4.18)$$

Substituting (4.15) and (4.18) into (4.17) gives

$$\begin{aligned} \dot{V} &= q_{ri} \left[ Y_i(q_{ai}, q_{vi}, q_i, \tau_{si}) \tilde{\theta}_i - K_{Di} q_{ri} - K_{ii}^{-1} B_i q_{ri} - \right. \\ &\quad \left. K_{ii}^{-1} \lambda_i M_i q_{ri} + U_f + K_{ii}^{-1} F_q(q_i, \dot{q}_i) + K_{ii}^{-1} \delta_i \right] \\ &\quad - q_{ri} Y_i(q_{ai}, q_{vi}, q_i, \tau_{si}) \Gamma_i^{-1} \Gamma_i \tilde{\theta}_i \\ &= -K_{Di} q_{ri}^2 - K_{ii}^{-1} B_i q_{ri}^2 - K_{ii}^{-1} \lambda_i M_i q_{ri}^2 + q_{ri} U_f \\ &\quad + q_{ri} K_{ii}^{-1} F_q(q_i, \dot{q}_i) + q_{ri} K_{ii}^{-1} \delta_i \end{aligned} \quad (4.19)$$

If  $|q_{ri}| \geq \varepsilon$ , combining (4.13) and (4.19) yields

$$\begin{aligned} \dot{V} &= -K_{Di} q_{ri}^2 - K_{ii}^{-1} B_i q_{ri}^2 - K_{ii}^{-1} \lambda_i M_i q_{ri}^2 \\ &\quad - |q_{ri}| \left( \rho - \left( K_{ii}^{-1} F_q(q_i, \dot{q}_i) + K_{ii}^{-1} \delta_i \right) \frac{q_{ri}}{|q_{ri}|} \right) \\ &< -K_{Di} q_{ri}^2 - K_{ii}^{-1} B_i q_{ri}^2 - K_{ii}^{-1} \lambda_i M_i q_{ri}^2 < 0. \end{aligned} \quad (4.20)$$

If  $|q_{ri}| < \varepsilon$ ,

$$\begin{aligned}
\dot{V} &= -K_{Di}q_{ri}^2 - K_{ii}^{-1}B_iq_{ri}^2 - K_{ii}^{-1}\lambda_iM_iq_{ri}^2 \\
&\quad + q_{ri}\left(-\rho\frac{q_{ri}}{\varepsilon} - K_{ii}^{-1}F_q(q_i, \dot{q}_i) - K_{ii}^{-1}\delta_i\right) \\
&\leq -K_{Di}q_{ri}^2 - K_{ii}^{-1}B_iq_{ri}^2 - K_{ii}^{-1}\lambda_iM_iq_{ri}^2 - \frac{\rho}{\varepsilon}q_{ri}^2 + \rho|q_{ri}|
\end{aligned} \tag{4.21}$$

Since the last two terms of (4.21) achieves a maximum value at  $|q_{ri}| = \varepsilon/2$ , we have

$$\dot{V} \leq -K_{Di}q_{ri}^2 - K_{ii}^{-1}B_iq_{ri}^2 - K_{ii}^{-1}\lambda_iM_iq_{ri}^2 + (\rho\varepsilon/4) \tag{4.22}$$

From (4.22), it can be concluded that a Lyapunov function can be found if only

$$|q_{ri}| > \sqrt{\rho\varepsilon / \left(4(K_{Di} + K_{ii}^{-1}B_i + K_{ii}^{-1}\lambda_iM_i)\right)}.$$

Define  $S = \left\{q_{ri} \in R^1 \mid q_{ri}^2 \leq (\rho\varepsilon) / \left(2(K_{Di} + K_{ii}^{-1}B_i + K_{ii}^{-1}\lambda_iM_i)\right)\right\}$

Then on the surface of  $S, \partial S$ , we have  $\dot{V} \leq -\rho\varepsilon/4$ .

Denote  $T$  as the time for the solution trajectory to intersect the surface  $\partial S$ , then  $V(q_{ri}(T)) - V(q_{ri}(0)) \leq -(\rho\varepsilon/4)T$ , and

$$T \leq 4(V(q_{ri}(T)) - V(q_{ri}(0))) / \rho\varepsilon$$

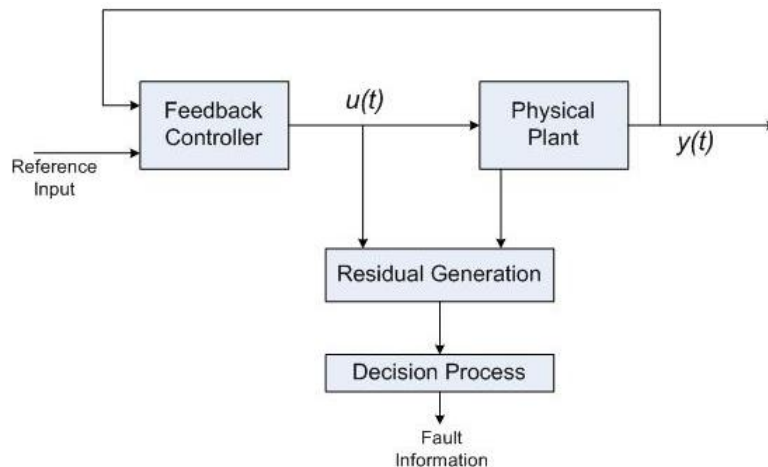
The boundedness of  $T$  is thus proved. And hence the uniform ultimate boundedness of  $q_{ri}$  is proven.

The boundedness of  $q_{ri}$  implies the boundedness of  $\tilde{q}_i$  and  $\dot{\tilde{q}}_i$ , since the definition of  $q_{ri}$  can be viewed as a stable first order differential equation in  $\tilde{q}_i$ , with  $q_{ri}$  as the input. Thus the boundedness of  $q_{ri}$  implies the boundedness of  $\tilde{q}_i$  and  $\dot{\tilde{q}}_i$  and therefore, of  $q_i$  and  $\dot{q}_i$  (Slotine and Li, 1991).

Since the controller guarantees the uniform ultimate boundedness, and the capability of the controller in handling faults that introduce larger errors are guaranteed. For any initial condition the tracking error is uniformly ultimately bounded. The tracking error approaches and stays within the bound in a finite time.

## 4.2 Fault Detection

In recent decades, the model based analytical redundancy method has received significant attention (survey papers by Frank 1990, Gertler 1988, Isermann 1984 and books by Patton *et al.* 1989, Gertler 1998 and Chen *et al.* 1999). The advantage of the analytical redundancy technique lies in the fact that the required redundancy is generated by powerful information processing approaches instead of additional physical hardware in the plant. The comparison between the system's actual measurement and its estimation generated by a mathematical model will generate the *residual* signal. These residual signals are normally zero or within threshold and become non-zero or exceed the threshold limits as a result of faults. The fault detection scheme based on analytical redundancy is as illustrated in Figure 4-2, where  $u(t)$  and  $y(t)$  are the system input and output respectively. The decision making process usually involves the selection of thresholds for the generated residuals. The threshold selection problem has been investigated in (Walker, 1989, Walker and Gai, 1979) and by calculating the minimum detectable fault in the frequency domain (Ding and Guo, 1998). The advantages of adaptive threshold over fixed thresholds have also been researched upon (Emami-Naeini *et al.*, 1988, Frank, 1997).



**Figure 4-2. The model-based analytical redundancy approach**

#### 4.2.1 Fault Detection Scheme

In the design of a fault detection method, it is assumed that no two faults occur at the same instant, which is a reasonable assumption normally made in reliability engineering. In this work it is assumed that only one fault occurs at each joint at a specific time instant. Then an actuator fault occurring at each joint can be found using an observer based scheme where actual joint velocity of each joint is compared against the estimated joint velocity.

The dynamical equation of the  $i^{th}$  joint module of a modular robot is given by

$$\begin{aligned} \tau_i = I_{mi} \gamma_i \ddot{q}_i + & \left( F_c + F_s \exp(-F_\tau \dot{q}_i^2) \right) \text{sat}(\dot{q}_i) \\ & + F_q(q_i, \dot{q}_i) + B\dot{q}_i + \delta_i + (\tau_{ji}/\gamma_i) \end{aligned} \quad (4.23)$$

The nominal values of friction model parameters can be estimated through offline techniques. The bound of velocity estimation error can be found during fault free operation of the system.

The velocity estimation error is defined as

$$e_i = v_i - \dot{q}_i \quad (4.24)$$

where  $v_i$  denotes the velocity estimate.

The velocity estimate can be obtained from the dynamical equation of the  $i^{th}$  joint given by (4.23) and the nonlinear observer proposed in (Xian, *et al.*, 2004) to guarantee error convergence in the presence of uncertainties. The following velocity estimate is obtained by integrating the acceleration term from expression (4.23):

$$v_i = \frac{1}{I_{mi} \gamma_i} \left( \int_0^t \tau_i dt - \int_0^t (\tau_{ji}/\gamma_i) dt - \int_0^t B\dot{q}_i dt \right) - L_i \quad (4.25)$$

and

$$\dot{v}_i = \frac{1}{I_{mi} \gamma_i} \left( \tau_i - (\tau_{ji}/\gamma_i) - B\dot{q}_i - \left( F_c + F_s \exp(-F_\tau \dot{q}_i^2) \right) \text{sat}(\dot{q}_i) \right) - \dot{L}_i \quad (4.26)$$

with the observer  $L_i$  given by

$$L_i = \int_0^t K_1 e_i(\sigma) d\sigma + \int_0^t K_2 \text{sat}(e_i(\sigma)) d\sigma - \int_0^t (U_i / I_{mi} \gamma_i) d\sigma \quad (4.27)$$

where

$$U_i = \begin{cases} -\rho_{Fi} (e_i / |e_i|) & |e_i| \geq \varepsilon_0 \\ -\rho_{Fi} (e_i / \varepsilon_0) & |e_i| < \varepsilon_0 \end{cases} \quad (4.28)$$

$K_1 > 0, K_2 > 0$  and  $\varepsilon_0$  is a positive parameter. The term  $U_i$  is used to compensate for the term  $\delta_i$  constituted by the effects of lower  $i-1$  joint modules. From (4.25) it is evident that to estimate velocity of each joint module, no information on other joints are required other than the bounds defined in (3.3) – (3.4). The term  $L_i$  is used to reduce the error between the estimated and measured velocity signals, and  $L_i$  by itself cannot be interpreted as a velocity observer.

Estimation error  $e_i$  obtained from estimated and measured velocities is used as the residue for fault detection with a threshold  $\varepsilon_{if}$ , which is a positive value obtained from fault free operation of the system.

A fault is declared if  $|e_i| > \varepsilon_{if}$ , i.e., the estimation error exceeds the selected threshold. The threshold can be set based on the various trials conducted in absence of faults. The time derivative of (4.24) is given by

$$\dot{e}_i = \dot{v}_i - \ddot{q}_i \quad (4.29)$$

$$\dot{e}_i = -K_1 e_i(t) - K_2 \text{sat}(e_i(t)) + \left( \frac{U_i + \delta_i + F_q(q_i, \dot{q}_i)}{I_{mi} \gamma_i} \right) \quad (4.30)$$

**Theorem 4.2:** *Given an  $n$ -DOF modular robot, with joint dynamics as given in (4.23) and an observer defined in (4.27), then for each joint, the velocity estimation error given by (4.24) is uniformly ultimately bounded during the fault free operation of the modular robot.*

**Proof:**

For stability analysis the Lyapunov function candidate is defined as

$$V = \frac{1}{2} e_i^2 \quad (4.31)$$

Differentiating (4.31) yields

$$\dot{V} = e_i \dot{e}_i \quad (4.32)$$

Substituting (4.30) into (4.32) yields

$$\dot{V} = -K_1 e_i^2 + \frac{e_i}{I_{mi} \gamma_i} \left( U_i + \delta_i + F_q(q_i, \dot{q}_i) \right) - e_i K_2 \text{sat}(e_i(t)) \quad (4.33)$$

Thus if  $|e_i| \geq \varepsilon_0$ , then using (4.28), the expression  $\dot{V}$  in (4.33) becomes

$$\begin{aligned} \dot{V} &= -K_1 e_i^2 - e_i K_2 \text{sat}(e_i(t)) \\ &\quad - \left( \rho_{Fi} / I_{mi} \gamma_i \right) |e_i| + \left( 1 / I_{mi} \gamma_i \right) \left( \delta_i + F_q(q_i, \dot{q}_i) \right) e_i \\ &= -K_1 e_i^2 - e_i K_2 \text{sat}(e_i(t)) \\ &\quad - \left( |e_i| / I_{mi} \gamma_i \right) \left( \rho_{Fi} - \left( \delta_i + F_q(q_i, \dot{q}_i) \right) \left( e_i / |e_i| \right) \right) < 0 \end{aligned}$$

If  $|e_i| < \varepsilon_0$ , then using (4.28), the expression  $\dot{V}$  in (4.33) becomes

$$\begin{aligned} \dot{V} &= -K_1 e_i^2 - e_i K_2 \text{sat}(e_i(t)) + \frac{e_i}{I_{mi} \gamma_i} \left( -\frac{\rho_{Fi} e_i}{\varepsilon_0} - \delta_i - F_q(q_i, \dot{q}_i) \right) \\ &\leq -K_1 e_i^2 - e_i K_2 \text{sat}(e_i(t)) + \left( 1 / I_{mi} \gamma_i \right) \left( -\rho_{Fi} \frac{e_i^2}{\varepsilon_0} + \rho_{Fi} |e_i| \right) \\ &\leq -K_1 e_i^2 - \frac{\rho_{Fi}}{I_{mi} \gamma_i} \frac{e_i^2}{\varepsilon_0} - \left( K_2 - \frac{\rho_{Fi}}{I_{mi} \gamma_i} \right) |e_i| \end{aligned}$$

The expression  $-\frac{\rho_{Fi}}{I_{mi} \gamma_i} \frac{e_i^2}{\varepsilon_0} - \left( K_2 - \frac{\rho_{Fi}}{I_{mi} \gamma_i} \right) |e_i|$  achieves a maximum of

$\left( \rho_{Fi} - K_2 I_{mi} \gamma_i \right)^2 \varepsilon_0 / \left( 4 \rho_{Fi} I_{mi} \gamma_i \right)$  at  $|e_i| = \left( \rho_{Fi} - K_2 I_{mi} \gamma_i \right) \varepsilon_0 / \left( 2 \rho_{Fi} \right)$ . Thus we have

$$\dot{V} \leq -K_1 e_i^2 + \left( \left( \rho_{Fi} - K_2 I_{mi} \gamma_i \right)^2 \varepsilon_0 / \left( 4 \rho_{Fi} I_{mi} \gamma_i \right) \right).$$

From the above expression it can be concluded that a Lyapunov function can be found if only

$$|e_i| > \sqrt{\left( \rho_{Fi} - K_2 I_{mi} \gamma_i \right)^2 \varepsilon_0 / \left( 4 K_1 \rho_{Fi} I_{mi} \gamma_i \right)}.$$

Define  $S = \left\{ e_i \in R^1 \mid e_i^2 \leq \left( \rho_{Fi} - K_2 I_{mi} \gamma_i \right)^2 \varepsilon_0 / \left( 2 K_1 \rho_{Fi} I_{mi} \gamma_i \right) \right\}$

Then on the surface of  $S$ ,  $\partial S$ , we have

$$\dot{V} \leq -\left(\left(\rho_{Fi} - K_2 I_{mi} \gamma_i\right)^2 \varepsilon_0 / \left(4\rho_{Fi} I_{mi} \gamma_i\right)\right).$$

Denote  $T$  as the time for the solution trajectory to intersect the surface  $\partial S$ , then

$$V(e_i(T)) - V(e_i(0)) \leq -\frac{\left(\rho_{Fi} - K_2 I_{mi} \gamma_i\right)^2 \varepsilon_0}{\left(4\rho_{Fi} I_{mi} \gamma_i\right)} T.$$

Hence,

$$T \leq \frac{\left(V(e_i(T)) - V(e_i(0))\right) \left(4\rho_{Fi} I_{mi} \gamma_i\right)}{\left(\rho_{Fi} - K_2 I_{mi} \gamma_i\right)^2 \varepsilon_0}. \text{ Thus the boundedness of the}$$

velocity estimation error  $e_i$  is proved.

### 4.3 Simulation Results

To study the effectiveness of the proposed fault control and fault detection scheme, a 3-DOF serial robot manipulator, working on a horizontal plane with the following parameters, is used for the simulations:

$$\begin{aligned} F_{ti} &= 100 s^2 / rad^2, B_i = 1.5 Nms / rad, F_{ci} = 3.5 Nm, F_{si} = 1 Nm, \\ K_{Di} &= 100, \kappa_i = 0.1, l_i = 0, \lambda_i = 100, \Gamma = 1 I^{6 \times 6}, \varepsilon = 0.1, \rho = 2, \rho_F = 1 \end{aligned} \quad (4.34)$$

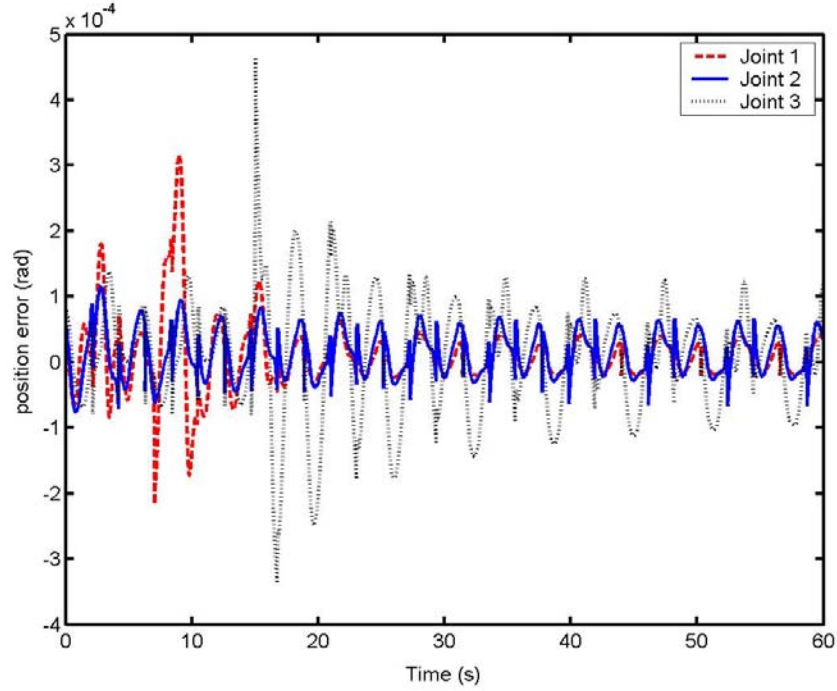
**Table 4-1. Parameters of the simulated system**

	Link 1	Link 2	Link 3
Mass of link (kg)	8	5	4
Length of link (m)	1	1	1
Link inertia (kg-m <sup>2</sup> )	1.0	0.8	0.6
Dist. to centre of mass (m)	0.5	0.5	0.5
Rotor inertia (kg-m <sup>2</sup> )	0.4	0.2	0.1
Gear reduction ratio	10	10	10

The dynamic equations of the model are as given in Appendix B. The nominal parameters of the friction model are assumed as

$$\hat{B}_i = 1.2 Nms / rad, \hat{F}_{ci} = 1 Nm, \hat{F}_{ti} = 80 s^2 / rad^2, \quad \hat{F}_{si} = 0.8 Nm, \hat{\kappa}_i = 0.2, \hat{l}_i = 0.5. \quad \text{For}$$

simplicity, the same friction model and parameters were considered for all the three joints. For the simulations, the parameters of the manipulator are chosen as given in Table 4-1.



**Figure 4-3. Tracking errors under actuator faults(at  $t = 7$  s on first joint and  $t = 15$  s on third joint)**

#### 4.3.1 Fault Tolerant Control

The desired trajectories for the three joints are selected as  $q_d = \sin(t) - 0.5\sin(2t)$  for  $0 \leq t \leq 60$  s. The following controller parameters are used for the simulations

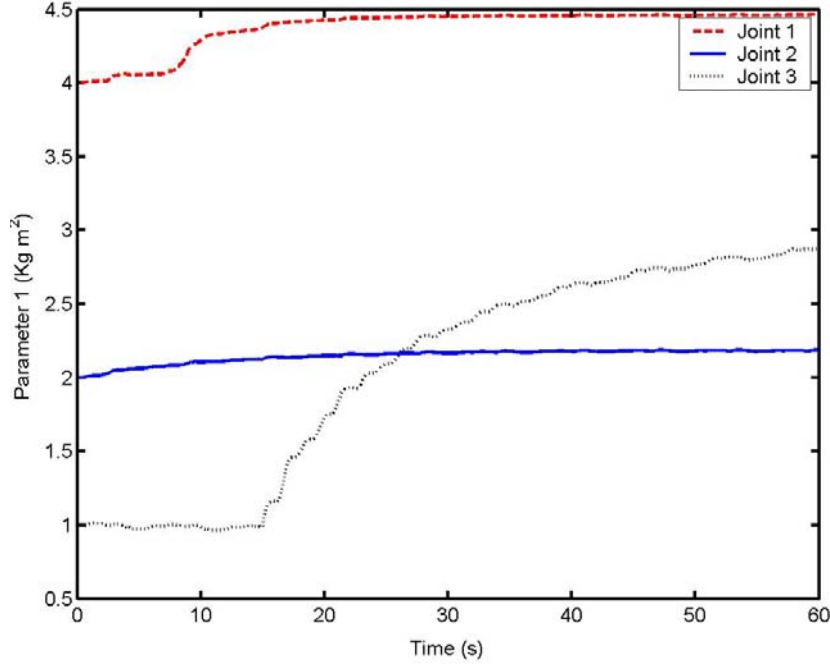
$$F_{\tau i} = 100 \text{ s}^2 / \text{rad}^2, B_i = 1.5 \text{ N ms} / \text{rad}, F_{ci} = 3.5 \text{ Nm}, F_{si} = 1 \text{ Nm},$$

$$K_{Di} = 100, \kappa_i = 0.1, l_i = 0, \lambda_i = 100, \Gamma = 1 I^{6 \times 6}, \varepsilon = 0.1, \rho = 2$$

The actuator effectiveness factor for first joint module was changed from unity to 0.7 at  $t = 7$  s and the corresponding value for third module was changed from unity to 0.4 at  $t = 15$  s. The occurrence of fault is detected by the fault detection algorithm and the fault is tolerated by the control algorithm. The fault tolerant control scheme ensures



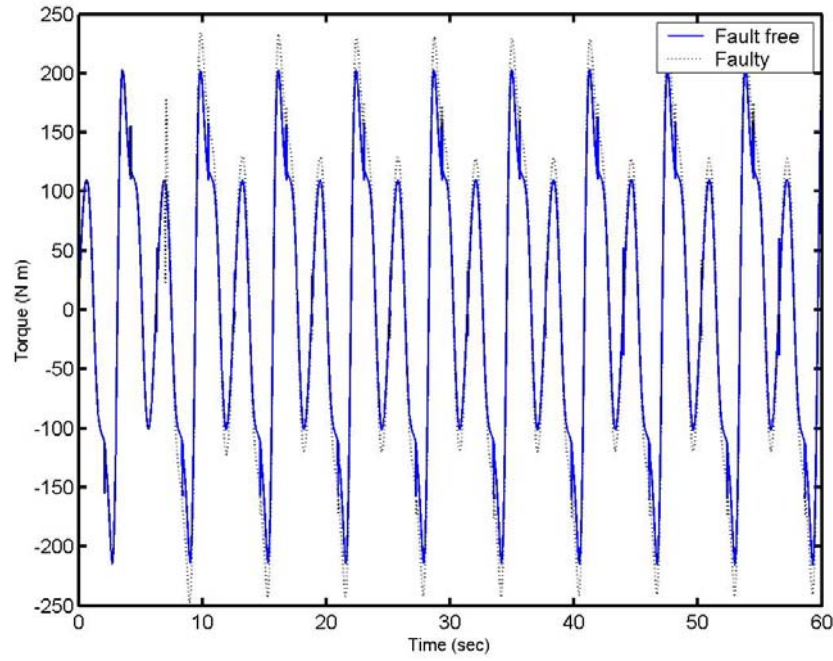
that the tracking error does not diverge even under a change in actuator effectiveness factor. This is depicted in Figure 4-3.



**Figure 4-4. Changes in parameter  $\theta_i^1$  under fault occurrence**

Figure 4-3 shows the tracking errors for all the three joints, under the above mentioned actuator faults. The variations in actuator effectiveness factors are evident in the appreciable tracking error changes for the first and third joint. But along time the tracking error reduces despite the deviation in actuator effectiveness factor.

The presence of actuator faults causes variations in the parameter vector  $\theta_i$ . The changes in parameter  $\theta_i^1$ , the first parameter of vector  $\theta_i$ , with the decrease in actuator effectiveness factor on the first and third joints, are shown in Figure 4-4. There are substantial changes in tracking error and parameter estimates as a result of the actuator degradation occurring at those specific time instants. The control torque on first joint during a fault-free and faulty operation is shown in Figure 4-5.

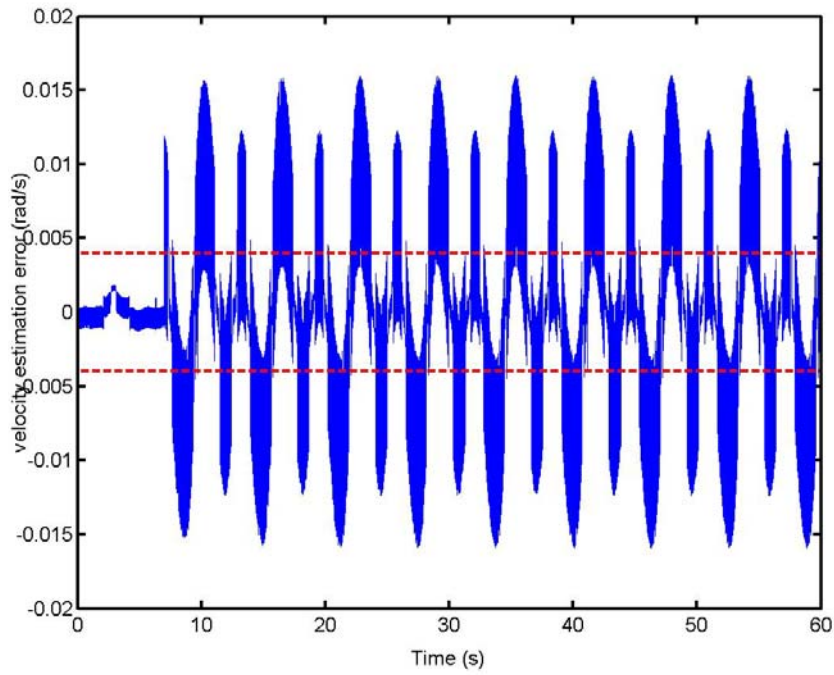


**Figure 4-5. Commanded torque at first joint**

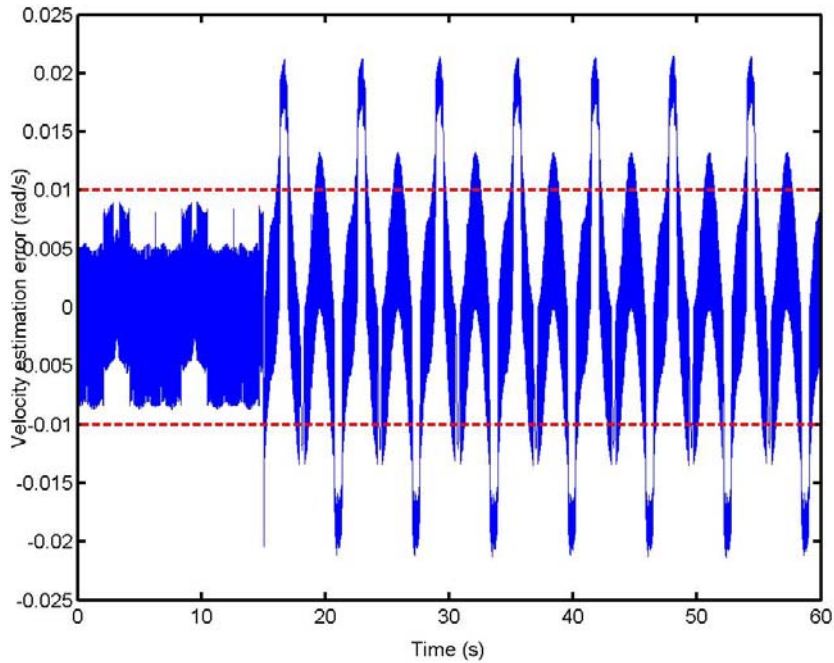
### 4.3.2 Fault Detection

For the 3-DOF modular robot, simulations were carried out to study the effectiveness of the fault detection technique. After various trials under absence of faults, the threshold values  $\varepsilon_{if}$  were chosen as 0.004, 0.008 and 0.01 for the first, second and third joints, respectively. The observer gains for all three joints were chosen as  $K_1 = 20$  and  $K_2 = 10$ .

The obtained results are as shown in Figure 4-6 and Figure 4-7, showing the velocity estimation errors of the first and third joint modules, respectively. As expected the velocity estimation error exceeds the pre-defined threshold values at  $t = 7s$  for the first joint and  $t = 15s$  for the third joint, as a result of the introduced change in their actuator effectiveness factors.



**Figure 4-6. Velocity estimation error on the first joint with actuator fault at  $t = 7$  s**



**Figure 4-7. Velocity estimation error on third joint, with actuator fault at  $t = 15$  s**

Though the effect of sensor dynamics does not adversely affect the proposed fault tolerant and fault detection scheme, it is worth noting the effects caused by sensor dynamics, namely, sensor delay and sensor ripples. At this juncture it must be pointed out that the effects of sensor delay would cause a delay in the fault detection scheme, *i.e.*, the velocity estimation error exceeds the threshold at a later time instant than the case with zero sensor delay. The overall velocity estimation error plot with time delay in torque sensor is similar to that in Figure 4-6, except that the presence of sensor delay would cause a delay in the detection scheme.

Depending on the priority of the task under execution and severity of the fault, a decision can be made to immediately abort the process without causing further damage or to finish the task in progress. To give an indication of the severity of fault at each joint, ‘health’ of each joint needs to be monitored continuously. The velocity estimation error indicates the severity of the deviation of actuator effectiveness factor. With larger deviations of effectiveness factor from unity, the velocity estimation error would correspondingly increase. The information could be used for the maintenance of the system.



## **Chapter 5**

### **Experiments**

The simulations have demonstrated the effectiveness of the proposed control schemes, and experiments were done to further substantiate the claims as presented in this chapter. The experiments have also demonstrated how applicable these control schemes were to practical applications.

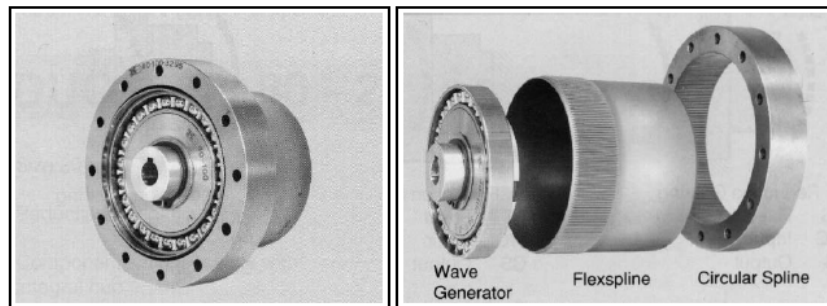
#### **5.1 Experimental Setup**

The experimental setup involves a 3-DOF modular and reconfigurable robot, and associated hardware and software, that have been developed in our laboratory. The mechanical components include three reconfigurable joints and mechanical links. The joint has several major components that will be outlined. The components will be divided into two major categories, the joint hardware and the joint software and communication. The following sections will give a brief description of each component.

Robotic manipulators require actuators with high torque capability at low velocities, and hence many electrically actuated robots use a gear transmission to increase the torque and decrease the drive speed. Among gear transmissions, harmonic drives have the advantage of being compact and light-weight, together with high gear ratio and minimal backlash.

Developed in 1955 primarily for aerospace applications, harmonic drives are high-ratio and compact torque transmission systems that have enjoyed wide industrial applications. This mechanical transmission employs a continuous deflection wave along a non-rigid gear to allow for gradual engagement of gear teeth. Because of this gear-tooth meshing action, harmonic drives can deliver high reduction ratios in a very small package.

Every harmonic drive consists of three components as illustrated in Figure 5-1. The wave generator is a ball bearing assembly with a rigid, elliptical inner race and a flexible outer race. The flexspline is a thin walled flexible cup with small, external gear teeth around its rim. The circular spline is a rigid ring with internal teeth machined along a slightly larger diameter than those of the flexspline. When assembled, the wave generator is nested inside the flexspline, causing the flexible part to adopt the elliptical profile of the wave generator, and the external teeth of the flexspline to mesh with the internal teeth of the circular spline.



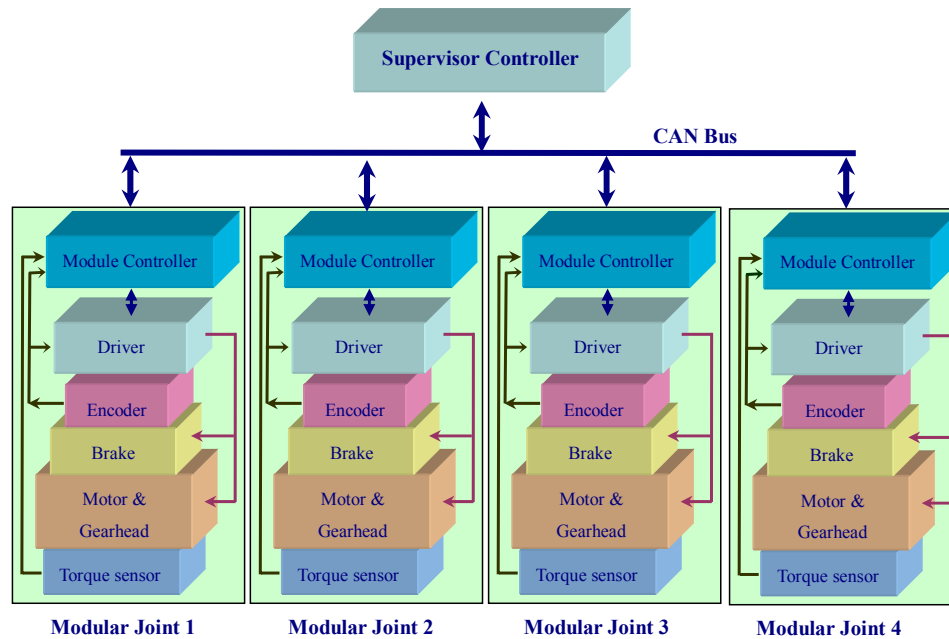
**Figure 5-1. The components of harmonic drive transmission**

Since the harmonic drive has three rotational ports, by using different combinations of rotations on these parts, numerous differential gearing functions and reduction ratios can be achieved. In this configuration, the circular spline is fixed and a low-torque, high-speed motor is used to drive the wave generator to produce a high-torque, low-speed rotation on the flexspline.

The harmonic drive displays superior performance features as compared to conventional gear transmissions (Taghirad, 1997). These performances are as noted:

- High torque capacity
- Light weight and compact
- Zero backlash
- High efficiency
- Nonlinearity

### 5.1.1 Control System Architecture



**Figure 5-2. Control system architecture for modular robot system**

The implemented control system architecture is shown in Figure 5-2. The CANOpen protocol is adopted on the supervisor, communication and module level. Each of the modular joints is comprised of DSP motion controller, motor drive, torque sensor, harmonic drive, position encoder, electric motor and associated circuitry.

### 5.1.2 Hardware

- Motor: Moog brushless DC motor model BN-42-33EU-03. The motor has a peak torque of 8.82 Nm, a rated speed of 4710 RPM, a rated torque of 1.42 Nm and consumes a rated power of 697W.
- Gear: HD Systems harmonic drive CFS-32-100-2A-GR-IV-SP-A1228. The gear has a ratio of 101:1, a rated torque of 137 Nm a max momentary torque of 647 Nm and a maximum input speed of 4800 RPM.
- Encoder: Torque Systems HS15-05/05-2000-0-04-T5-01. The encoder has a resolution of up to 5000 counts per revolution. The input voltage is 5 to 26 volts DC with a frequency of 500 KHz.



- Power Supply: Sorenson model DCS 80-37E. The power supply is capable of a voltage of 0-80, and amperage of 0-37. The input: 190-250 VAC, three phase, 14A typical, 47-63 Hz.
- Joint Torque Sensor: The strain gauges are from Kyowa Electronic Instruments model KFG-1-120-D16-11. The gauges are 1mm long with a resistance of 120 $\Omega$ . The gauges have an excitation of +/- 5V and can produces up to 20mV output.
- Amplifier: Phoenix Systems, model PN5603007. Strain Gage amplifier supports strain gages/load cells with resistances from 120 to 20,000 ohms. It includes 1000 V, 3-way galvanic isolation between power supply, input, and output circuits. Module filters and conditions signals to eliminate unwanted signal noise with DIP switch selectable cutoff frequencies of 30 or 5000 Hz. Outputs are provided for voltage (0 to 10, +/-10, 0 to 5, or +/-5 V), and current loop (4 to 20 mA) operation.
- Link: The links are 34 cm long and designed to carry a payload or the electronics of the succeeding joints in order to create a torque signal to feedback to the controller to test to see if the control law is effective.

### 5.1.3 Electrical System Architecture

The overall electrical system architecture is shown in Figure 5-3. The personal computer (PC) is used to program the digital signal processor (DSP) board, which is a distributed controller. Once the program is downloaded into the DSP board, it is capable of controlling the driver, which drives the motor. The motor turns the gear head and produces two feedbacks, an encoder for position as well as the torque sensor. The encoder is a digital signal and is sent to the driver. The driver uses the position signal to calculate the velocity. These two pieces of information are fed back to the DSP board via a CAN bus which is a high speed communication network device. The DSP also receives the analog signal from the amplifier which is necessary to boost the weak torque signal into the +/-3V range that the DSP board needs to do the analog to digital

conversion. The DSP board now has the torque signal, the position and velocity signals and is able to calculate how much torque to send to the driver.

#### **5.1.3.1 Driver**

The driver selected for this experimental setup is from Elmo Motion Control. The drive is a Cello digital servo drive. The driver is capable of doing position, velocity and current control. This unit was selected because the current loop is open, so it is possible to program a current control law and implement it with this driver. The drive has a standard serial port, RS232 and a Controller Area Network (CAN) communication port available. For the purposes of the MRR project, only the CAN communications are necessary.

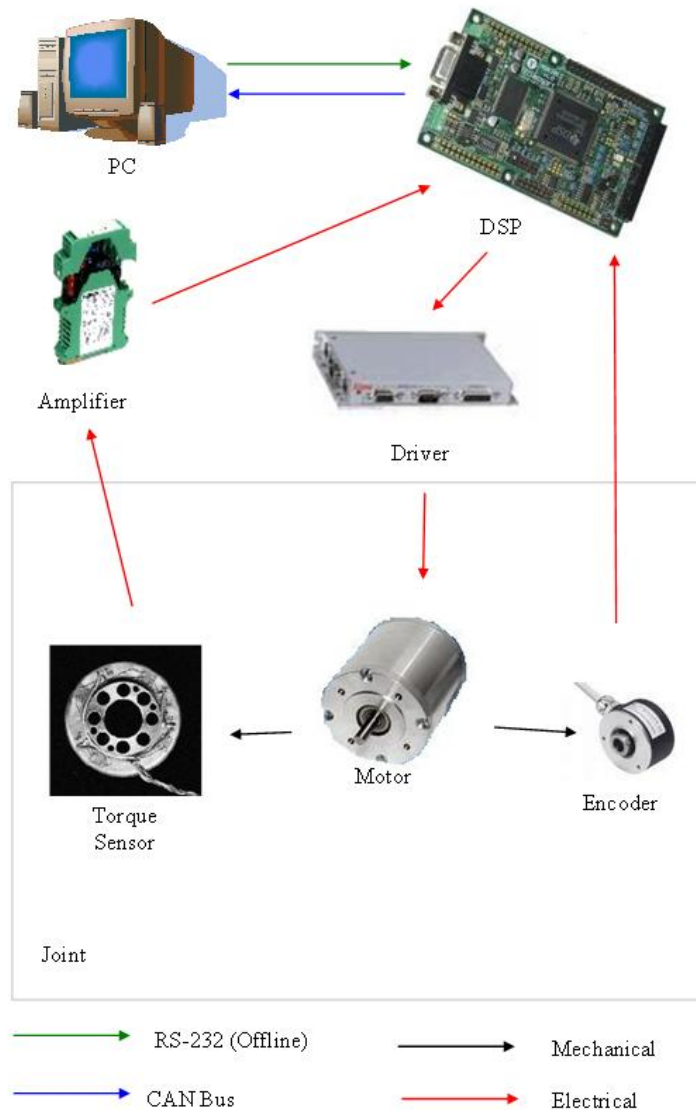
#### **5.1.3.2 DSP Board**

The DSP board serves as a distributed controller. For the MRR project, there would be a DSP board on each joint, which would be capable of controlling each joint in a distributed manner. The DSP is responsible for controlling the joint in real-time. The communication from the PC to the DSP board is carried out through the RS232 COM port. Technosoft provides software that is capable of programming the control law which is desired onto the DSP board. The software programming language is C.

#### **5.1.3.3 CANopen Protocol**

In order to ensure timely control, it is necessary to have high speed communication between the DSP board and the driver. This is accomplished by using the CAN bus and a CANopen protocol. The CANopen protocol has one master, in this case the DSP board, and any number of slave nodes. For the purposes of this experiment, the only slave node needed is the motor drive. The overall communication architecture is shown in Figure 5-3. The CANopen bus speed can be set as high as 1 Mbps, for a single joint this speed is not necessary. The speed that was used was 500 kbps. The limiting factor in determining the speed at which the joint is commanded was determined by the driver. The driver is designed to be a position control but had an option to keep the torque loop open. This allows for a control law to be executed outside of the driver, which is desirable for the research on the MRR. The drawback of this feature is that the drive

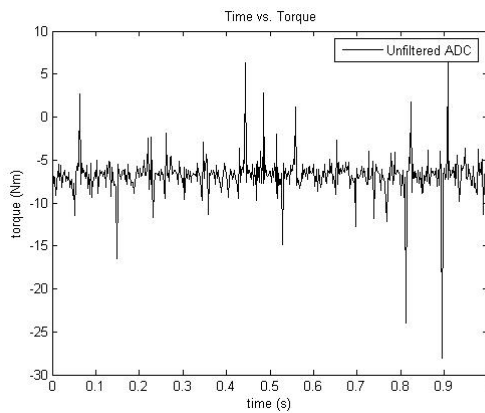
only reads the CAN bus torque commands in its processor idle loop. The processor idle loop is statistically read every 1.5 ms. While it is possible to broadcast torque commands more frequently, based on the bus speed, the DSP speed, it will not ensure that the drive will execute this command. Therefore the control law is set to send a command to the drive every 2 ms. This is the shortest amount of time which guarantees a consistent execution.



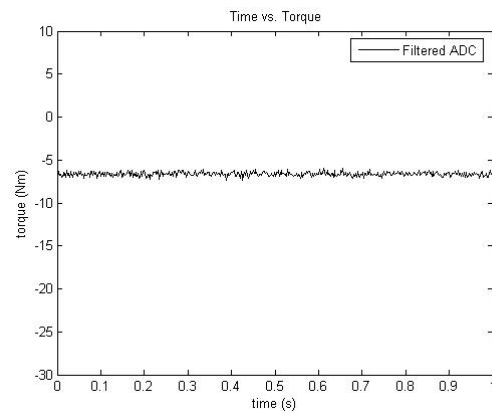
**Figure 5-3. Electrical communication**

#### 5.1.3.4 Sensors

Two sensors are used in the experiment. There is the joint torque sensor, and the position sensor, i.e., the incremental encoder. The incremental encoder is used for two purposes. The first is to measure the position. This procedure is fairly straightforward. The encoder is electrically wired to the DSP and there is a subroutine on the Technosoft DSP board that increments (or decrements depending on the direction of rotation) a register as the motor shaft rotates. The torque sensor is an analog signal. In order for it to be used in the control algorithm, it must be converted to a digital signal. The conversion from analog to digital is also a subroutine supplied by the Technosoft DSP board. The board is capable of converting sixteen different channels. Since this application requires only one channel, the other 15 channels are used to over sample the signal. Over sampling the signal and averaging the result dramatically reduces the analog to digital conversion (ADC) noise as seen in Figure 5-4 and Figure 5-5.



**Figure 5-4. Unfiltered ADC signal**



**Figure 5-5. Filtered ADC signal**

#### 5.1.3.5 Velocity Estimation

The control law requires the velocity of the joint to be known. A tachometer can be used to directly measure the velocity, but there is not one available in this experimental setup. In order to determine the velocity the signal from the position encoder is differentiated at each time step. At very low speeds this method can be

inaccurate. The definition of very low speed depends on the resolution of the encoder. When only a few pulses are being measured in each time step, then partial increments of the encoder are lost and the result is a loss of accuracy in velocity measurement.

Liu *et al.* (2004) proposed a method to estimate the velocity to overcome this problem. In the proposed method, if the encoder does not increment by a set number of pulses, or the time does not change by a pre-determined amount, then the velocity will remain the same. The resulting velocity is experimentally demonstrated to be more accurate than by directly computing the velocity at each time step. The method also allows the tuning of the estimation parameters in order to trade off the accuracy vs. the time delay.

#### **5.1.4 Mechanical Setup**

The 3-DOF modular robot is shown in Figure 5-6. Each joint is designed such that it can be rotated by 90 degrees. This is part of the reconfigurable characteristics of the joint. The link was designed to be able to be easily removed and reconfigured to two positions; the position shown in Figure 5-6 as well as a configuration shown in Figure 5-7. In the second position the weight at the end because of other joints and payload will give the maximum amount of torque on the base joint. In all of the experiments conducted for this work, the joint was configured exactly as seen in Figure 5-6 or Figure 5-7.

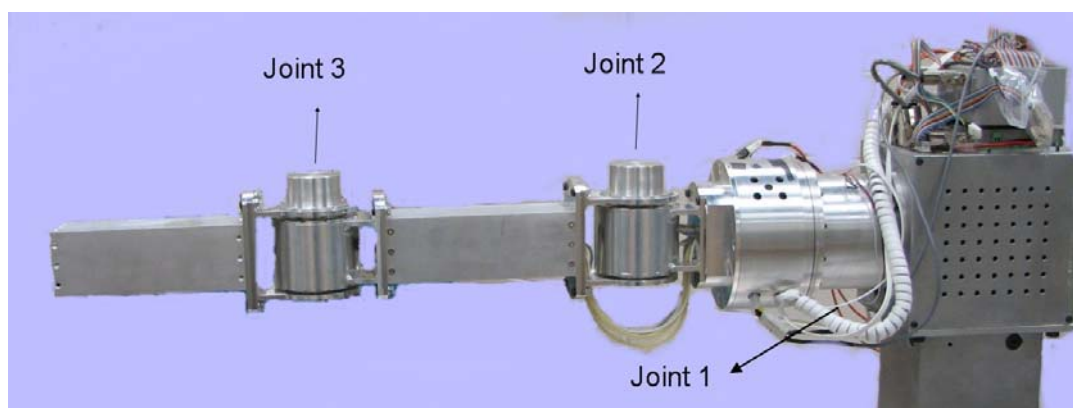
#### **5.1.5 Software**

The software components include the interfacing programs, DSP programs, controller implementation, communication software, etc.

For transfer of data from motion controller to the supervisory computer, a logger module working on CAN was implemented. The logged data can be retrieved at the end of each experiment. The flow charts of the software modules can be found in Appendix C.



**Figure 5-6. A modular and reconfigurable robot system – configuration 1**



**Figure 5-7. Modular and reconfigurable robot - configuration 2**

**Table 5-1. Parameters of the system**

	Link 1	Link 2	Link 3
$I_{mi}$	2.5e-4	3.46e-5	3.46e-5
$\lambda_i$	85	85	70
$\rho_{Fi}$	0.01	0.02	0.02
${}^i\mathcal{E}$	0.05	0.1	0.1
$k_i$	0.0005	0.0001	0.001
$B_i$	0.006	0.00124	0.00124
$F_{ci}$	0.2558	0.025	0.025
$F_{\tau i}$	15	15	15
$F_{si}$	0.34	0.027	0.027

## 5.2 Results for Torque Sensor based Control

The 3-DOF modular robot is configured as shown in the Figure 5-6. The distributed control law given in (3.16) is applied to this system represented by (2.1)-(2.3). The reference trajectory for the first joint was chosen as

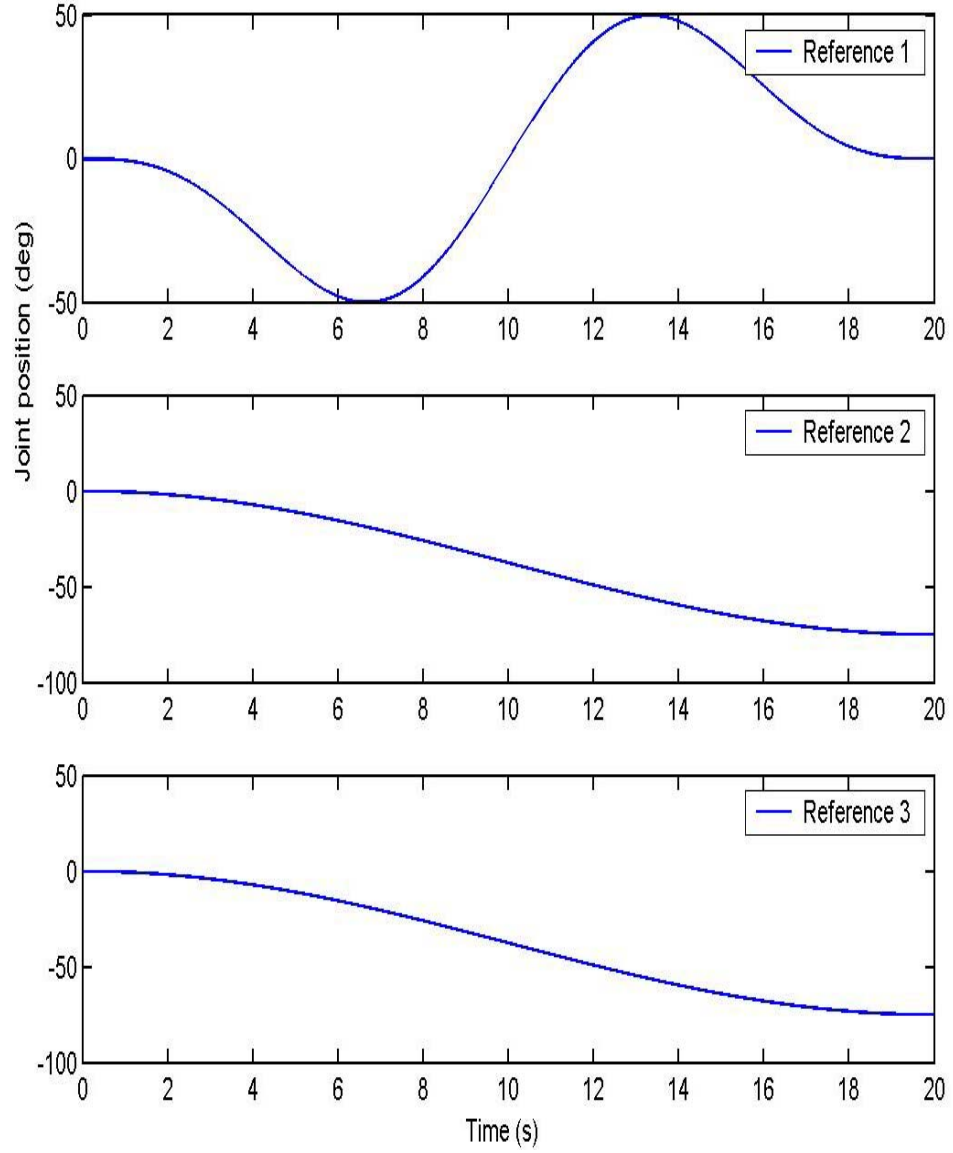
$$q_{d1} = A(\sin(4t/2\pi) - 0.5\sin(8t/2\pi)) \text{ for } 0 \leq t \leq 20\text{s}.$$

The amplitude of the reference trajectory was  $A = 2\pi/9\text{rad}$ . And for second and third joints the trajectories were chosen as

$$q_d = \frac{A}{2} \left( 1 - \cos \left( \pi \left( \frac{2t-1}{40} \right) \right) \right) \text{ for } 0 \leq t \leq 20\text{s}.$$

Amplitude of reference trajectories for second and third joints were chosen to as  $A = 5\pi/12\text{rad}$ . The reference trajectories are as shown in Figure 5-8.

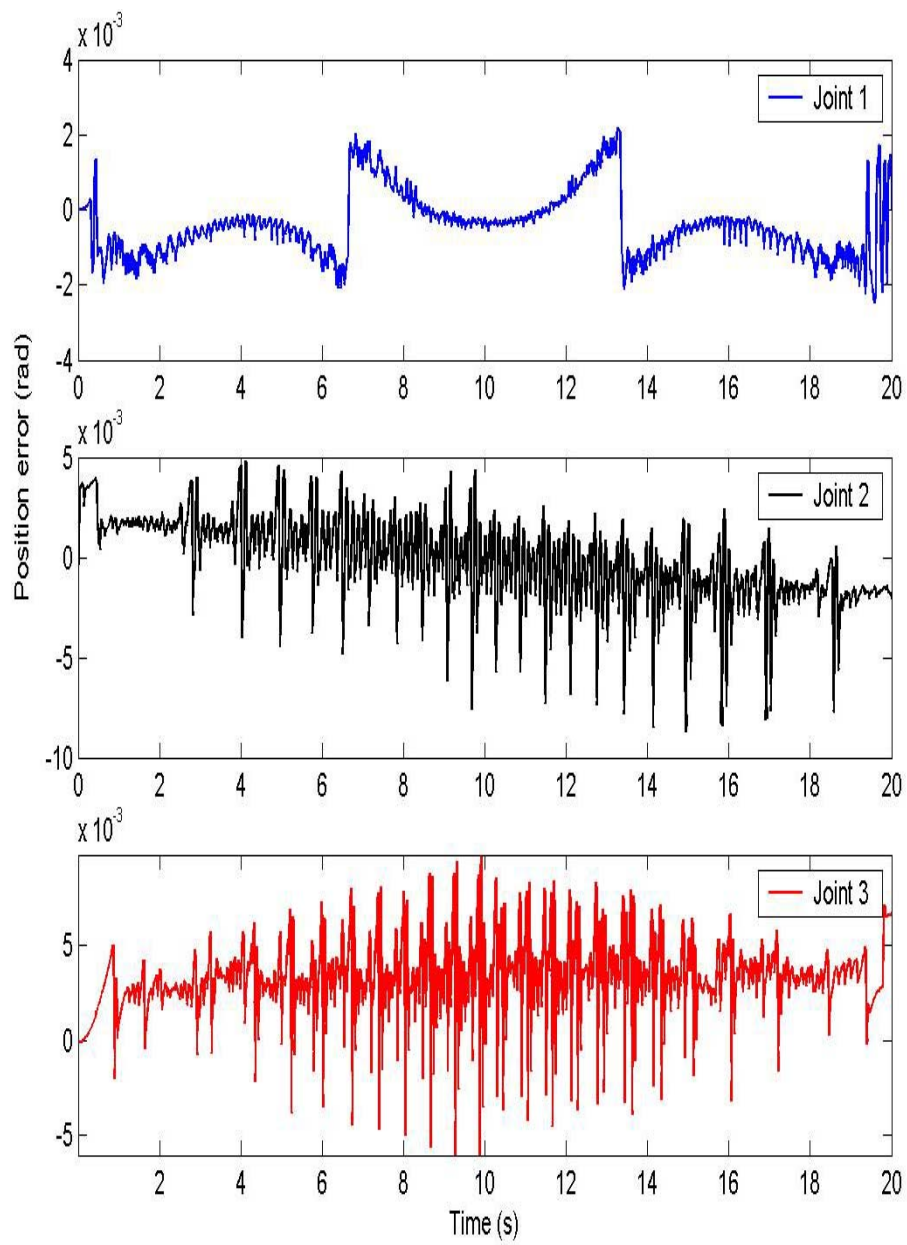
The controller parameters were chosen as in Table 5-1.



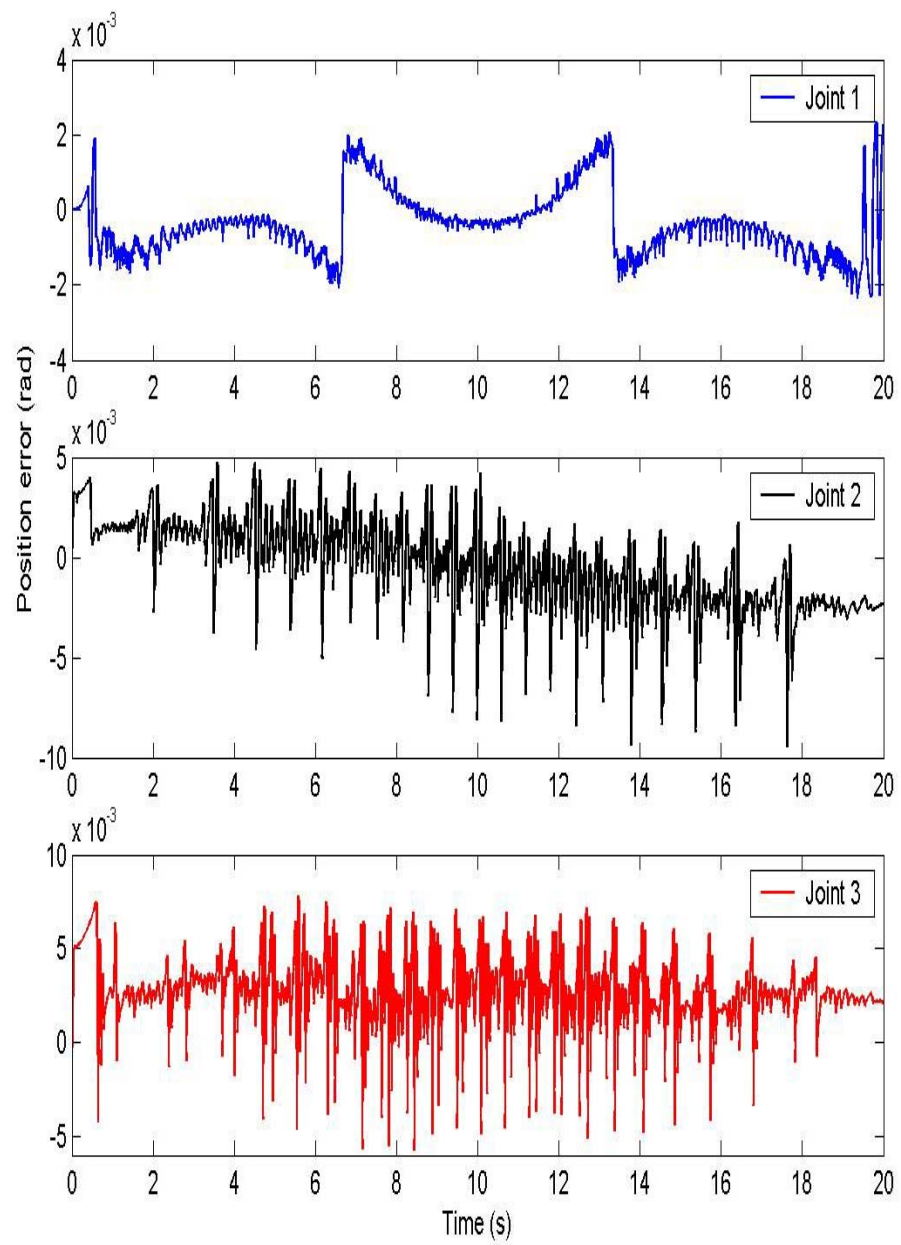
**Figure 5-8. Trajectories for the three joints**

The output plots for the robot configured, as in Figure 5-6, are as shown in Figure 5-9 to Figure 5-12. The tracking errors at the three joints are as shown in Figure 5-9 - Figure 5-11. Three different cases are studied: with no payload, with payload of 10N from beginning of the trial, and payload added at  $t = 7s$  (approx).

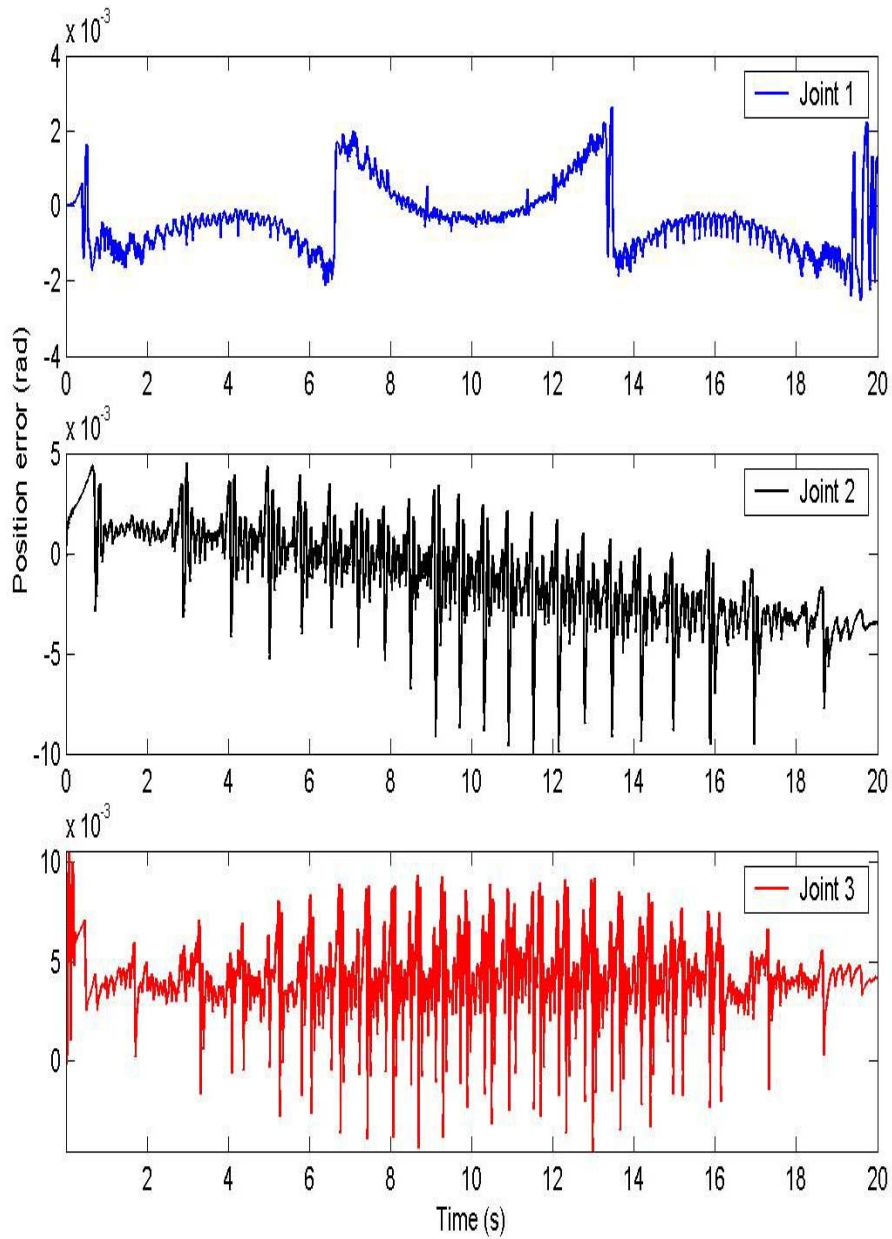




**Figure 5-9. Position tracking error at three joints with MRR in Configuration 1 (No payload)**



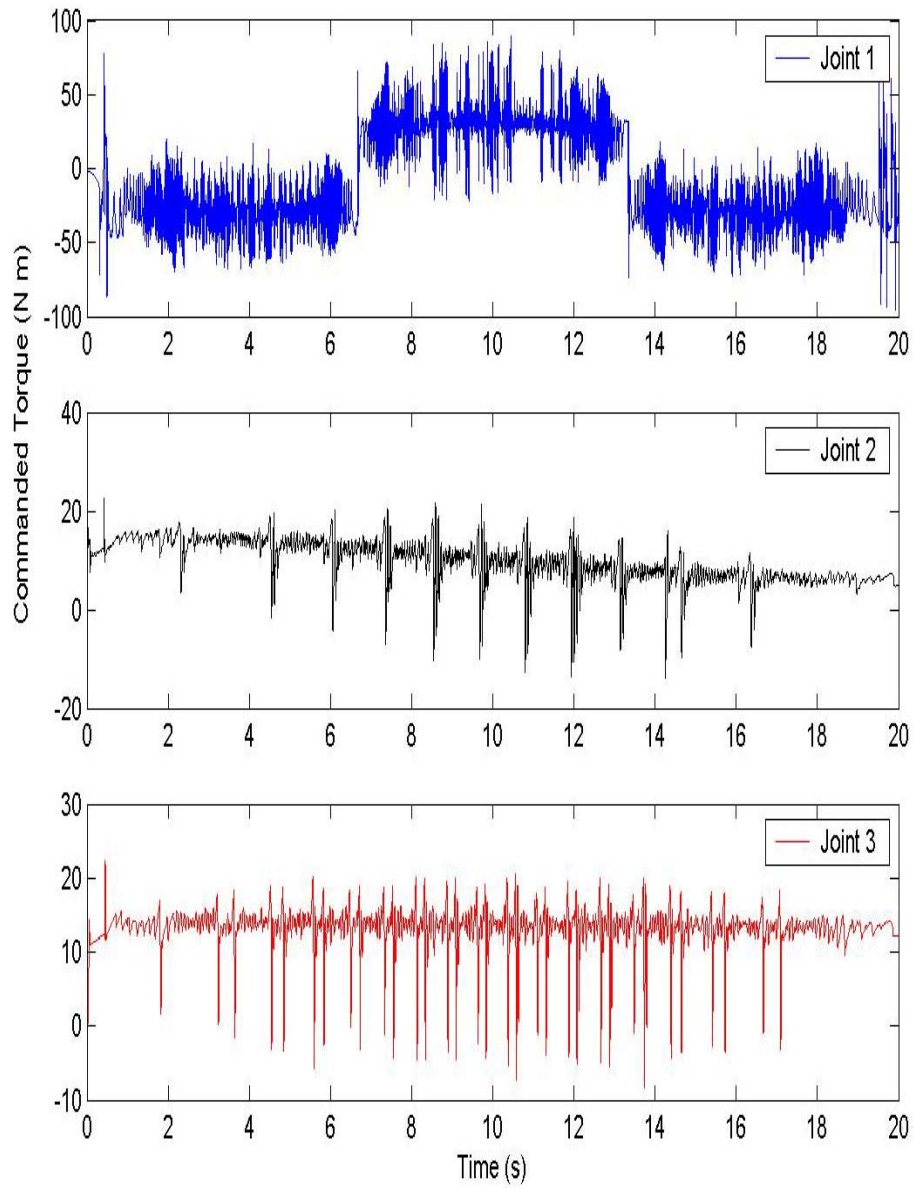
**Figure 5-10. Position tracking error at three joints (MRR in Configuration 1, payload added at  $t=7$ s)**



**Figure 5-11. Position tracking error at three joints with MRR in Configuration 1 (payload at  $t = 0$  s)**

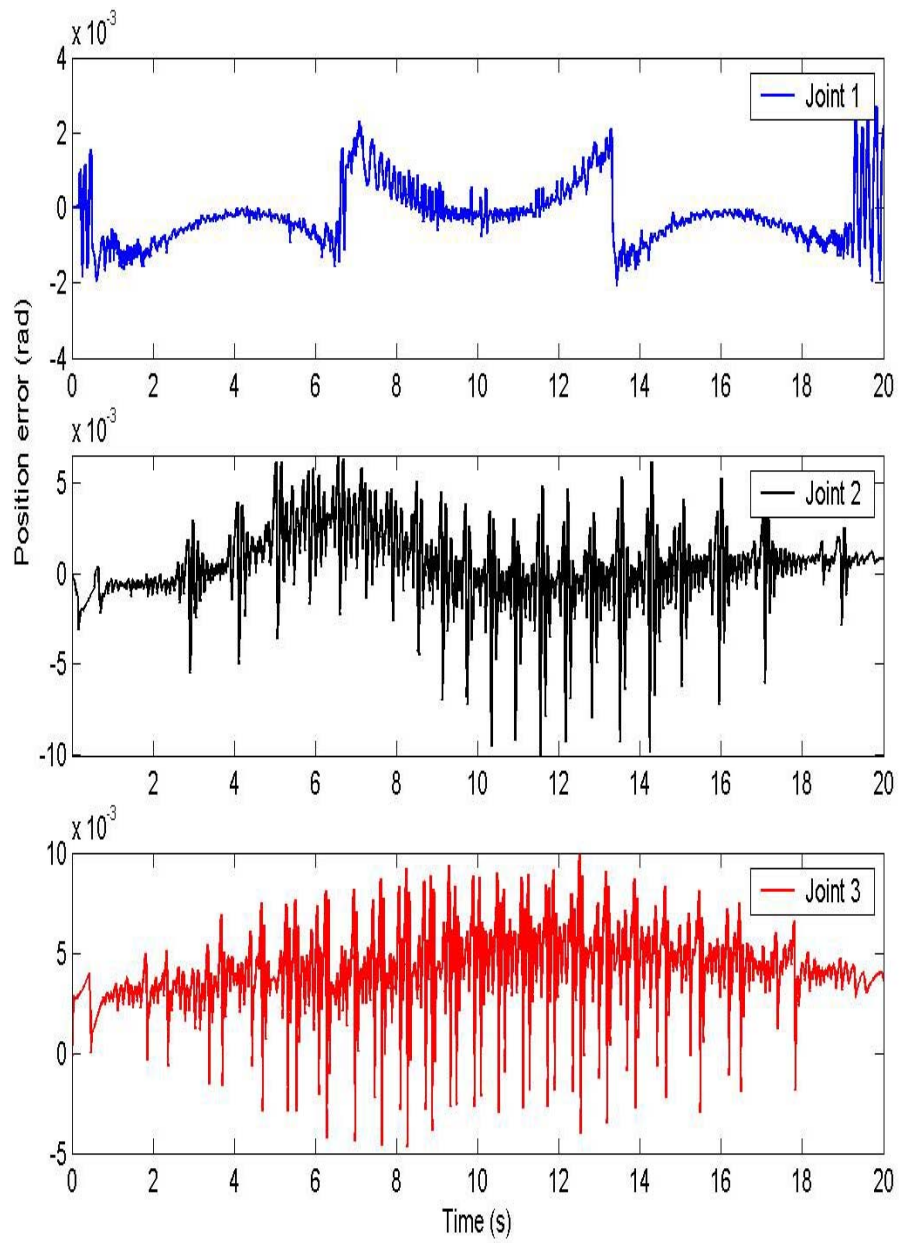
From the above plots it is evident that addition of payloads does not cause deterioration in tracking errors. Hence the effectiveness of controllers in quickly adapting to payload changes is proved.

The commanded torque at the three joints, for the case with no payload, is as shown in Figure 5-12.

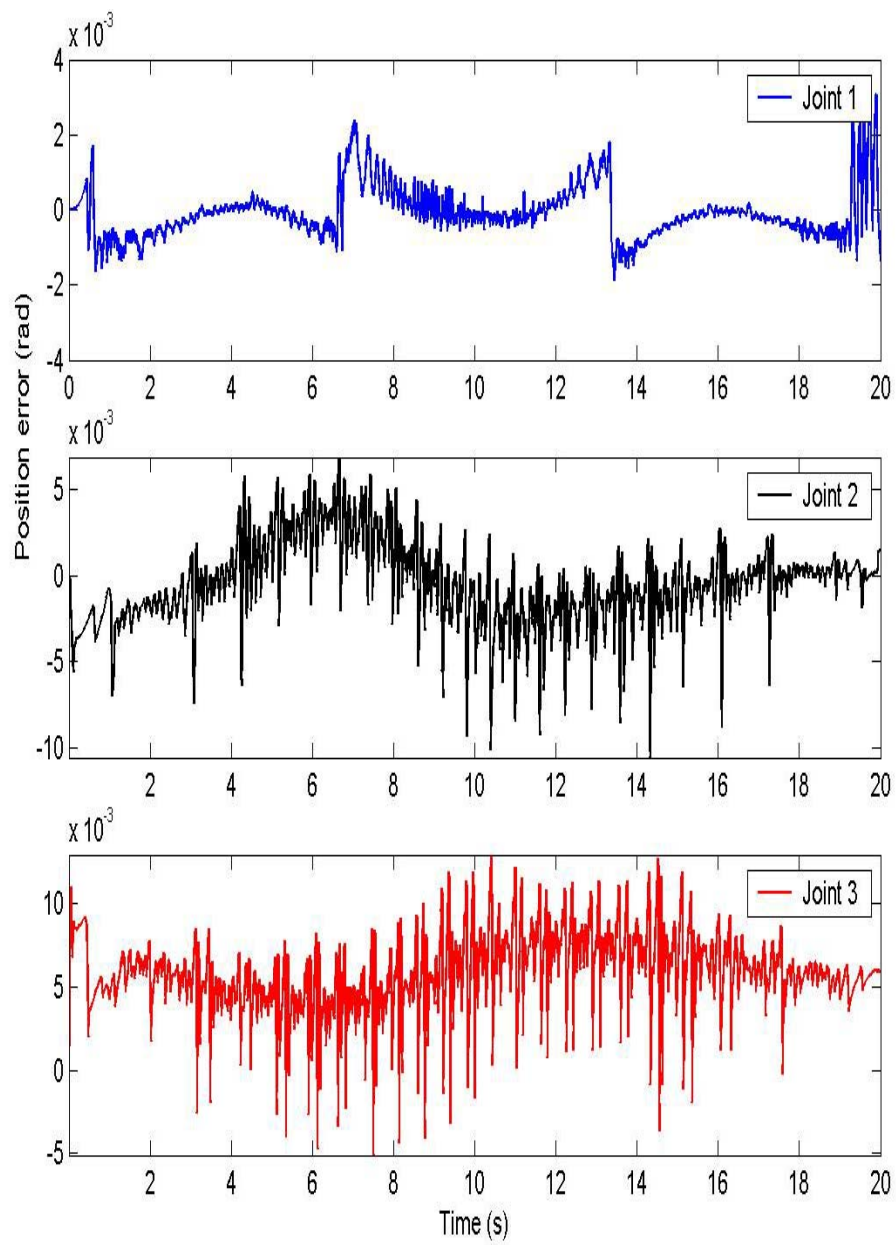


**Figure 5-12. Commanded torque at three joints of MRR system (Configuration 1)**

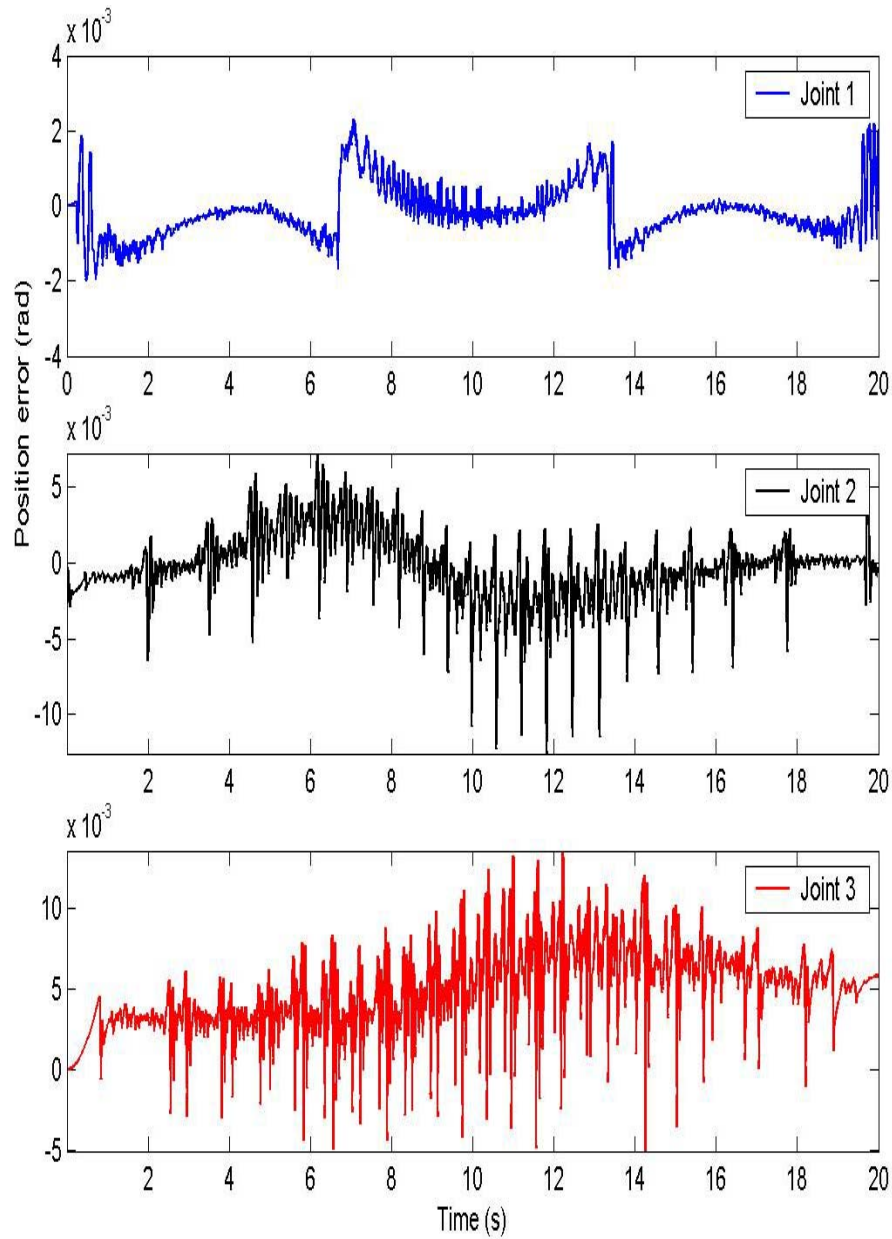
The same controller was used for tracking control of MRR, configured as in Figure 5-7. The output plots for the three cases considered are as in Figure 5-13 - Figure 5-15.



**Figure 5-13. Position tracking error at three joints, with no payload (MRR in configuration 2)**



**Figure 5-14. Position tracking error at three joints with payload at  $t = 0$  s (MRR in configuration 2)**



**Figure 5-15. Position tracking error at three joints with payload added at  $t = 7$  s (MRR in configuration 2)**

From Figure 5-9 - Figure 5-15, it can be seen that the tracking performance are quite similar for MRR working in either configurations, with or without payload. The controller is adaptive to various configurations of MRR, and controller re-tuning was not necessary to achieve the tracking performance.



### 5.3 Results for Decentralized Fault Tolerant Control

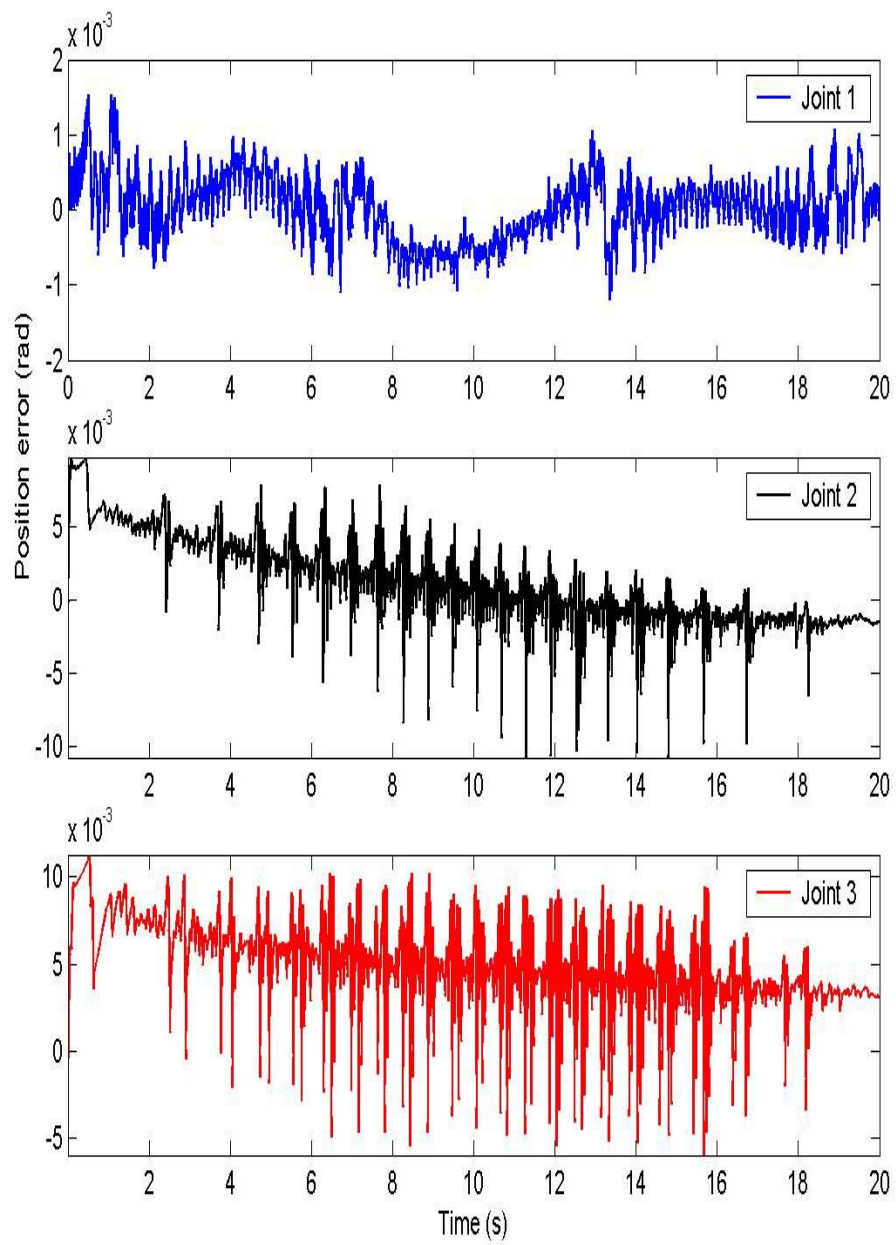
The desired trajectory of the joints are selected as

$$q_d = A(\sin(8t/2\pi) - 0.5\sin(16t/2\pi)) \text{ for } 0 \leq t \leq 20\text{s}.$$

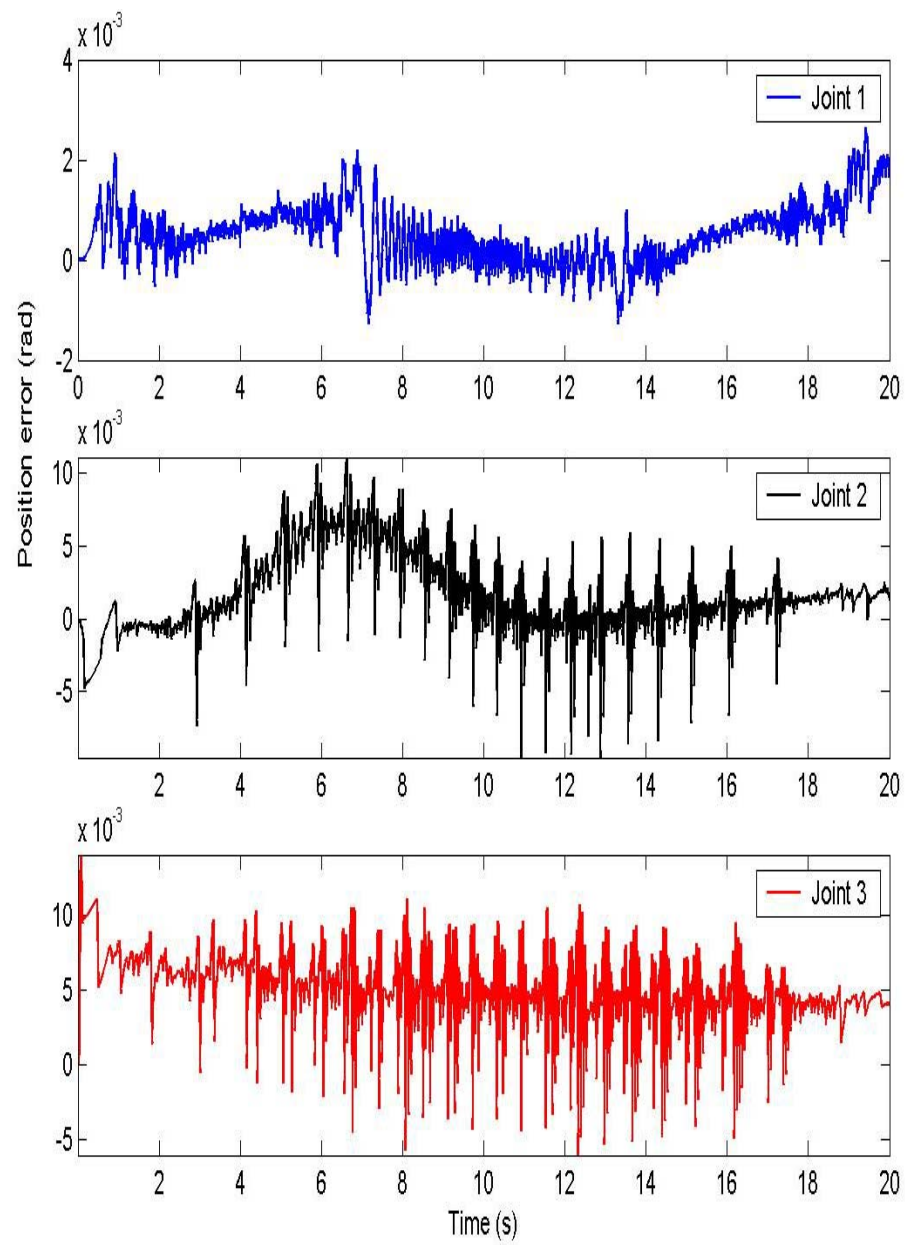
The amplitude of the reference trajectory  $A=0.2 \text{ rad}$ . The nominal parameters are chosen as  $\hat{k}_i=0.9$ ,  $\hat{F}_{ci}=0.2 \text{ Nm}$ ,  $\hat{B}_i=0.002 \text{ Nms/rad}$ ,  $\hat{F}_{ti}=0.05 \text{ s}^2/\text{rad}^2$ ,  $\hat{F}_{si}=0.1 \text{ s}^2/\text{rad}^2$  and  $\hat{l}_i=0.1$ . The motor rotor inertia is obtained from the data sheets as  $0.00025 \text{ Kg m}^2$  and the gear ratio of the harmonic drive is  $\gamma=101$ . The controller parameters are chosen as  $K_{Di}=0.05$ ,  $\lambda_i=40$ ,  $\Gamma=200 I^{6 \times 6}$ ,  $\rho=0.4$ ,  $\varepsilon=0.4$ ,  $\varepsilon_0=0.1$  and  $\rho_{Fi}=0.1$ . The observer gains were chosen as  $K_1=30$  and  $K_2=20$ .

The experiments for fault tolerant control are conducted for two different cases: fault free operation and a fault occurring at  $t=4\text{s}$  and  $t=6\text{s}$ . From the fault free operations the threshold value for velocity estimation error was fixed to be  $0.09 \text{ rad/s}$  and  $0.02 \text{ rad/s}$  for first and third joints respectively. The tracking error obtained under the fault free operation for both the MRR configurations are as shown in Figure 5-16 - Figure 5-17. The actuator fault is introduced by changing the actuator effectiveness factor from unity to  $0.5$  at  $t=4\text{s}$  for first joint and from unity to  $0.3$  at  $t=6\text{s}$  for third joint. The experiments are conducted for MRR in both configuration 1 and configuration 2. The obtained results are as shown in Figure 5-18 – Figure 5-23.

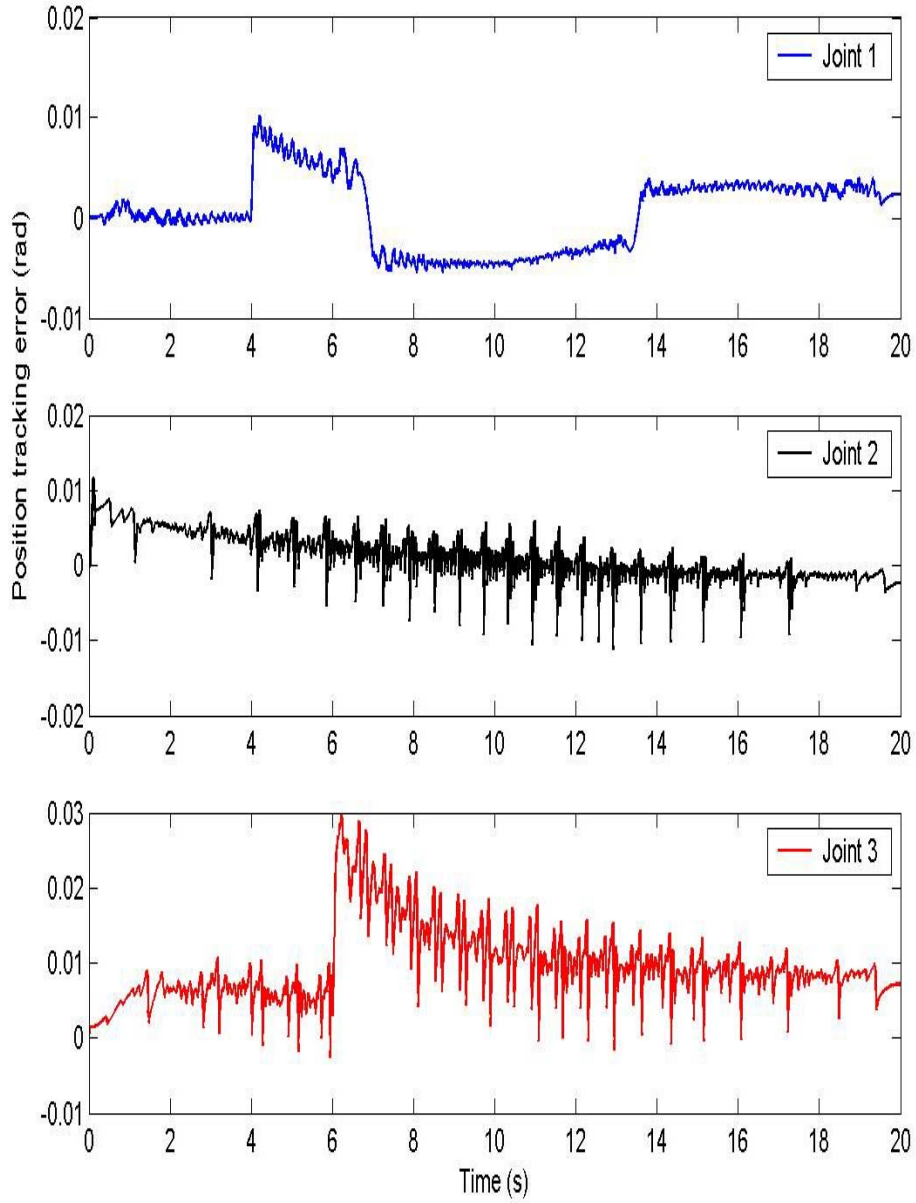




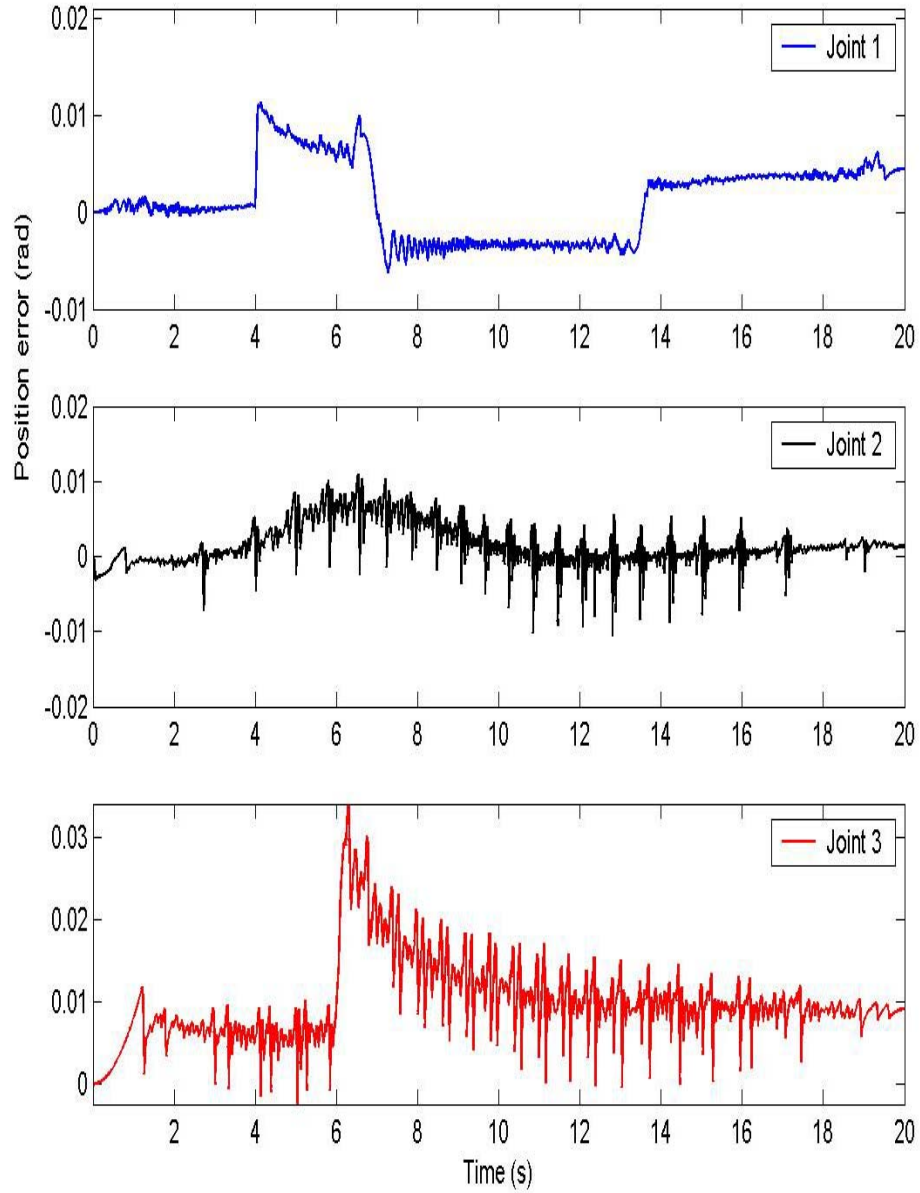
**Figure 5-16. Joint position error at three joints (fault free operation, MRR in Configuration 1)**



**Figure 5-17. Joint position error at three joints (fault free operation, MRR in Configuration2)**



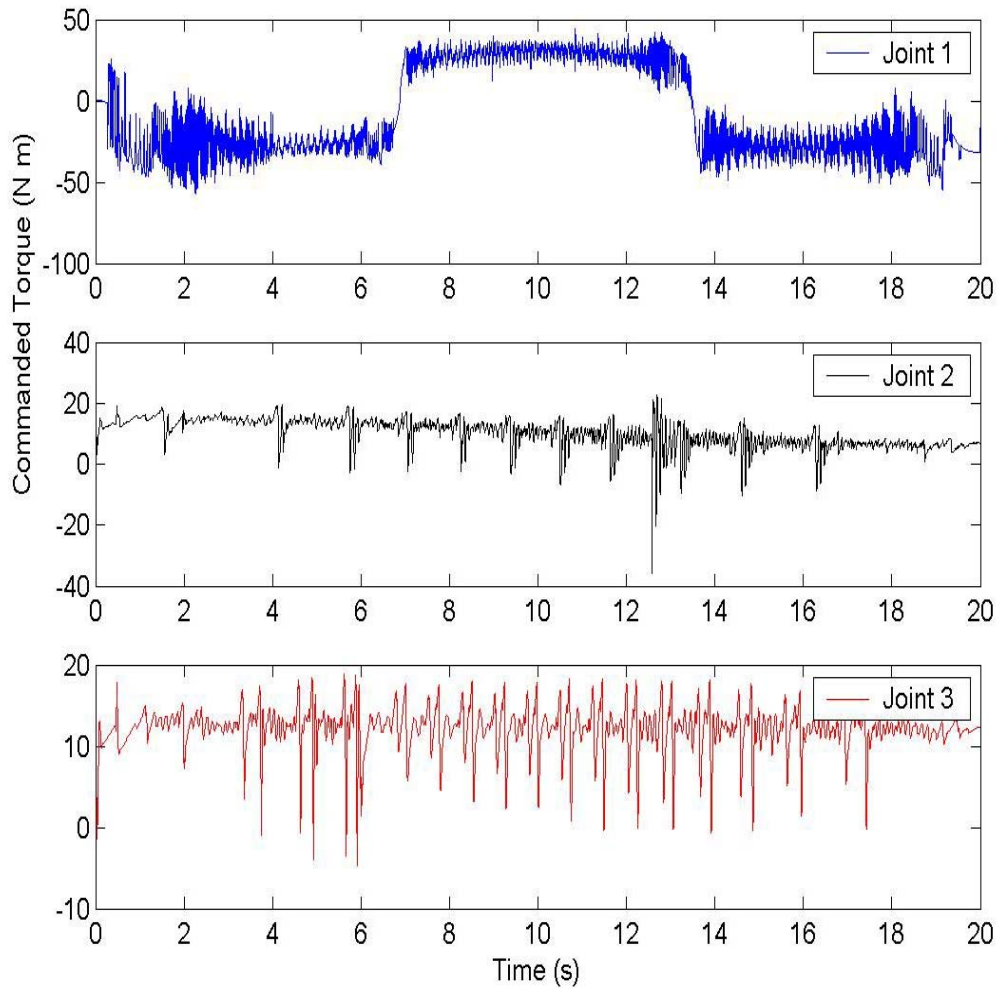
**Figure 5-18. Joint position error at three joints (fault occurrence at  $t = 4$  s and  $t = 6$  s, MRR in Configuration 1)**



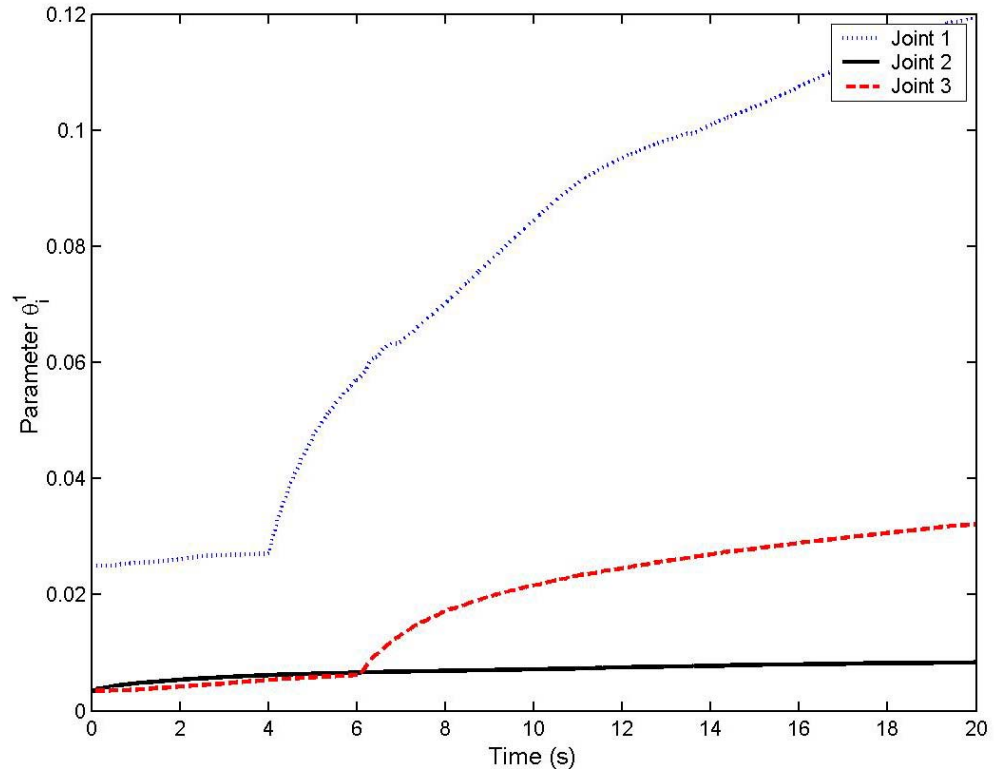
**Figure 5-19. Joint position error at three joints (fault occurrence at  $t = 4$  s and  $t = 6$  s, MRR in Configuration 2)**

In Figure 5-18 and Figure 5-19, the tracking errors occurring due to change in actuator effectiveness factor is shown. The tracking error shows a change at the instance of fault occurrence. It is seen that despite the change in actuator effectiveness factor, the

action of the fault tolerant control law ensures the position tracking error does not diverge. This guarantees the performance of the system does not deteriorate due to the changes in actuator effectiveness factor.



**Figure 5-20. Commanded torque for three joints (MRR in Configuration 1)**



**Figure 5-21. Changes in parameter  $\theta_i^1$  under fault occurrence**

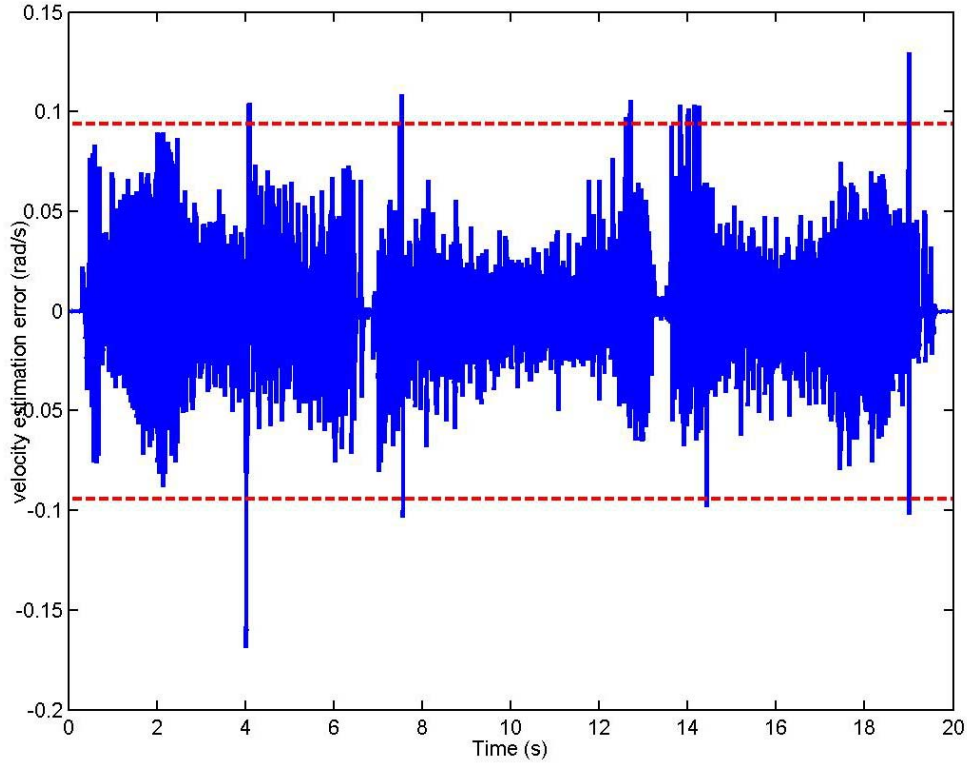
In Figure 5-20 and Figure 5-21, the commanded torque signal for all the three joints are shown, with MRR in configuration 1, and changes in the first parameter  $\theta_i^1$  in vector  $\theta_i$ . The variations in the parameter vector with decrease in actuator effectiveness factor can be seen in the above plot.

Though both the changes in payload and actuator effectiveness factor can result in parameter deviations and changes in tracking error, the payload changes are clearly distinguishable when used along with the torque sensor readings. The payload changes are recorded by the changes in torque sensor readings. Thus actuator faults can be easily differentiated from the payload changes.

## 5.4 Results for Fault Detection Scheme

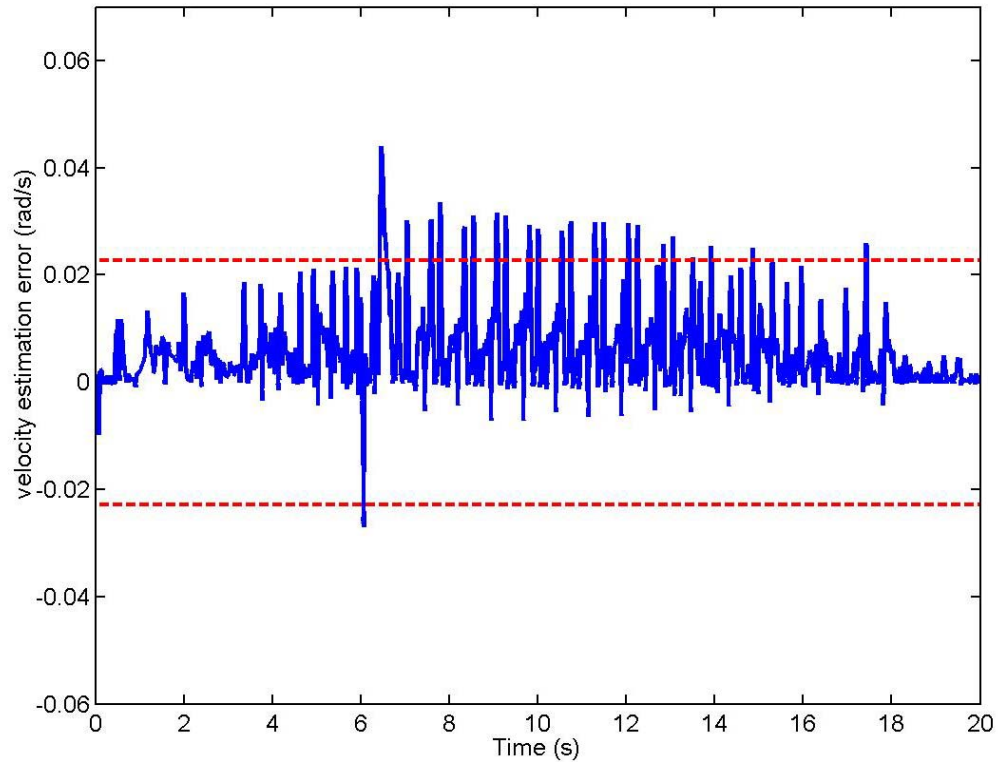
In Figure 5-22 and Figure 5-23, the velocity estimation errors occurring at first and third joints due to change in actuator effectiveness factor is shown. The velocity

estimation error shows a significant change at the instance of fault occurrence. The actuator fault occurrence at  $t = 4\text{s}$  and  $t = 6\text{s}$  causes the velocity estimation error to exceed the chosen threshold value. The data are collected for MRR in configuration 1.



**Figure 5-22. Velocity estimation error with actuator fault at first joint (fault occurrence at  $t = 4\text{ s}$ )**

The tracking error plots obtained using the two controllers can be compared, the distributed controller given by (3.16) and the fault tolerant controller given by (4.11) working under a fault free state. The error plots seen in Figure 5-9 and Figure 5-13 are seen to be quite comparable to the results in Figure 5-16 and Figure 5-17 obtained from the fault tolerant controller. The essential difference between the two controllers would be the robust approach used in controller (3.16) compared to the fault tolerant controller in (4.11) which uses an adaptive law.



**Figure 5-23. Velocity estimation error with actuator fault at third joint (fault occurrence at  $t = 6$  s)**

## 5.5 Discussions

A distributed control technique based on joint torque sensing is used for the control of modular and reconfigurable robot. The use of joint torque sensors offers a much quicker adaptation to payload changes, and hence better performance during such scenarios. The controller tracking errors were found to be within acceptable limits under payload variations. The tracking performance was studied for different configurations of the modular robot. The modularity offered by the control technique allowed reconfiguration of the 3-DOF robot, without controller re-tuning to achieve good tracking performance. The tracking performance of the modular robots under reconfigurations and payload variations, demonstrated the effectiveness of the distributed control strategy for the control modular and reconfigurable robots.

An actuator fault tolerant control method and a fault detection scheme for modular and reconfigurable robot with joint torque sensing have been implemented. Fault



tolerance and fault detection at each joint module are carried out independently of the other joints, *i.e.*, the controller at each joint does not require motion states of the other modules. This modular approach enables fault tolerant control and fault detection at individual modules, without affecting the control of other joints. Since the fault tolerant control is independent of the fault detection scheme, any delay caused in detection scheme does not affect the fault tolerant control action. The actuator effectiveness factors, torque sensor gains and offsets are incorporated into a parametric dynamical model formulation for each joint module of the robot. Based on this model formulation, a control scheme is designed for compensating the parametric uncertainties including the actuator effectiveness factors, torque sensor gains and offsets. Experimental results have confirmed the effectiveness of the proposed fault tolerant control and fault detection schemes.

## **Chapter 6**

### **Conclusions and Future Work**

#### **6.1 Conclusions**

The purpose of this thesis was to design and implement modular control algorithm, ideally suited for the control of modular and reconfigurable robots. The thesis also covered under its preview, a modular fault tolerant and fault detection strategy that can detect and compensate faults at module level. The effectiveness of the techniques was demonstrated on a 3-DOF modular and reconfigurable robot system.

A joint torque sensor based robot control avoids the requirement of link dynamics in the design of control law, and hence lessens the complexity of the control methodology. In addition the control law did not require the motion states of other joints. The distributed control law was ideal for a modular robot setup due to its joint-by-joint control nature. Simulations verified the robustness of the controller against torque sensor dynamics.

A modular fault tolerant control method was designed based on the joint-by-joint control technique. This enabled the fault tolerance to be achievable at each module, independent of the other modules. The use of joint torque sensor in the design methodology eliminated the requirement of link dynamics for the control, and hence made the control law computationally simpler. Since the torque sensor gains and offsets can change with temperature and other factors, uncalibrated torque sensor signals are used in the control law. A fault detection scheme was devised based on an error signal obtained from measured velocity and an estimated velocity signal. A threshold based procedure was used to determine the occurrence of faults. Though the velocity signal was sensitive to noise, the fault detection scheme still gave good results.

A comparison of the two control approaches showed that a comparable tracking performance is obtained under both techniques. The performance of the distributed

control law was compared to the fault tolerant controller under a fault free operating condition. While both controllers are operating under a distributed control strategy, one controller uses a decomposition based controller to achieve tracking performance; the other controller incorporates an added fault tolerant capability using a parameterization based on actuator effectiveness factor.

The experimental result shows the presence of high frequency signals in the output waveforms. This is mainly due to the vibration setting in from the base joint of the robot. But the good tracking performance obtained confirms the effectiveness of the proposed control laws, even under the effect of vibrations at the base joint.

The following conclusions could be drawn based on this work:

- Employing a joint torque sensor provides a new solution to the distributed control of modular and reconfigurable robots
- Joint by joint stabilization allows addition and removal of modules without the need for controller retuning
- In the discussed control law, the dynamic coupling and unmodeled dynamics of the system are compensated using decomposition-based robust controller
- The joint-by-joint control mentioned above enables a new modular approach for fault tolerance of modular and reconfigurable robots.
- The fault tolerant control law discussed allows fault tolerance at each module to be achieved independent of other modules.
- Since the fault tolerance and fault detection scheme is based on the above mentioned joint-by-joint control approach, the schemes do not require computation of link dynamics.
- The parameterization of torque sensor signals based on gain and offset allows the use of uncalibrated torque sensor signals in the control law.

## **6.2 Future Work**

A more complete model of the joint can be used in design of controllers. The model of harmonic drive and study on its flexibility will improve the performance of the

joint. The assumption on the rigidity of the joint can be removed to obtain a more precise mathematical model of the joint. This would involve the use of flexible joint robot model and study on vibrations at the joints too. Tracking control can be improved with a better system model, comprising of the above mentioned parameters.

The proposed fault tolerant control law guarantees the error convergence but not the parameter convergence. An adaptive scheme that can guarantee parameter convergence would give a better indication of the system during a faulty state. The variations in parameter could indicate fault occurrence and can be used for fault detection if parameter convergence can be achieved. In particular, an accurately estimated actuator effectiveness factor will indicate the health of the actuator.

The proposed fault detection scheme in Chapter 4 could be expanded to include detection of position and torque sensor faults. The present work only takes into consideration the actuator faults and not sensor faults.



## Bibliography

- Abdul, S., and Liu, G., (2007). "Fault tolerant control of modular and reconfigurable robot with joint torque sensing," *IEEE International Conference on Robotics and Biomimetics*, Sanya, China, pp. 1236-1241.
- Abdul, S., and Liu, G., (2008a). "Decentralised fault tolerant control and fault detection of modular and reconfigurable robot with joint torque sensing," *IEEE International Conference on Robotics and Automation*, pp. 3520-3526, California, USA.
- Abdul, S., and Liu, G., (2008b). "Decentralised fault tolerance and fault detection of modular and reconfigurable robot with joint torque sensing," *IEEE Trans. on Control Systems Technology*. (Accepted with revision conditions)
- Aghili, F., Buehler, M., and Hollerbach, J. M., (2001). "Design of a hollow hexa form torque sensor for robot joints," *Int. Journal of Robotics Research*, vol. 20, no. 12, pp. 967–976.
- Aghili, F., and Namvar, M., (2006). "Adaptive control of manipulators using uncalibrated joint-torque sensing," *IEEE Trans. on Robotics*, vol. 22, no.4, pp. 854-860.
- Armstrong-Helouvry, B., Dupont, P., and Canudas de Wit, C., (1994). "A survey of models, analysis tools and compensation methods for the control of machines with friction," *Automatica*, Vol.30, No.7, pp. 1083-138.
- Caccavale, F., and Walker, I. D., (1997). "Observer-based fault detection for robot manipulators," *Proc. IEEE Conf. Robotics and Automation*, pp. 2881-2887.
- Caccavale, F., (1998). "Experiments of observer-based fault detection for an industrial robot," *Proc. IEEE Int. Conf. on Control Applications*, pp. 480-484.
- Castano, A., Will, P., (2000). "Mechanical design of a module for reconfigurable robots", *Proc. of the 2000 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.2203-2209.

- Chen, W., Chen, I-Ming., and Wang, T., (1999). "Kinematic fault tolerant control for redundant robot based on joint velocities redistribution," *Int. Conf. Advanced Robotics*, pp. 397-402.
- Chen, W., Chen, I-Ming., Chen, C. Y., Yang, G., and Kang, I. -G., (1998). "Fault tolerant control for redundant robot based on joint torques redistribution," *Int. Conf. Control, Automation, Robotics and Vision*, pp. 1325-1329.
- Chen, J., and Patton, R. J., (1999). "*Robust model-based fault diagnosis for dynamic systems*," Kluwer Academic Publishers, London.
- Chirikjian, G., (1994). "Kinematics of a metamorphic robotic system," *Proc. IEEE Int. Conf. Robotics and Automation*, Vol. 1, pp. 449-55.
- Chladek, J. T., (1990). "Fault tolerance for space based manipulator mechanisms and control system," *First International Symposium on Measurement and Control in Robotics*, Houston, Texas.
- Craig, J. J., (1989). *Introduction to robotics: mechanical and control*, 2nd ed. Reading, Massachusetts: Addison-Wesley Publishing Company.
- De Luca, A., and Mattone, R., (2003). "Actuator failure detection and isolation using generalized momenta," *Proc. IEEE Int. Conf. on Robotics and Automation*, Taipei, Taiwan, pp. 634-639.
- De Luca, A., and Mattone, R., (2004). "An adapt-and-detect actuator FDI scheme for robot manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, New Orleans, LA, pp. 4975-4980.
- De Persis, C., and Isidori, A., (2001). "A geometric approach to nonlinear fault detection and isolation," *IEEE Trans. Automatic Control*, vol. 46, no.6, pp. 853-865.
- Ding, X., and Guo, L., (1998). "An approach to time domain optimization of observer based fault detection systems," *International Journal of Control*, vol. 69, no.3, pp. 419-442.
- Dixon, W. E., Walker, I. D., Dawson, D. M., and Hartranft, J. P., (2000). "Fault

detection for robot manipulators with parametric uncertainty: a prediction error based approach,” *IEEE Trans. on Robotics and Automation*, vol. 16, no. 6, pp. 689-699.

- Emami-Naeini, A., Akhter, M. M., and Rock, S. M., (1988). “Effect of model uncertainty on failure detection: the threshold selector,” *IEEE Trans. Automatic Control*, vol. 33, pp.1106-1115.
- Filaretov, V. F., Vukobratovic, M. K, and Zhirabok, A.N., (1999). “Observer-based fault diagnosis in manipulation robots,” *Mechatronics*, vol. 9, pp. 929-939.
- Filaretov, V. F., Vukobratovic, M. K, and Zhirabok, A.N., (2003). “Parity relation approach to fault diagnosis in manipulation robots,” *Mechatronics*, vol. 13, pp. 141-152.
- Frank, P. M., (1990). “Fault diagnosis in dynamic systems using analytical and knowledge based redundancy- a survey and some new results,” *Automatica*, vol. 26, pp. 459-474.
- Frank, P. M., (1997). “Survey of robust residual generation and evaluation methods in observer based fault detection systems,” *IFAC Journal of Process Control*, vol.7, pp. 403-424.
- Frank, D. M., Ding S. X., and Koppen-Selinger, B., (2000). “Current developments in the theory of FDI,” *Proc. IFAC Safeprocess*, Budapest, Hungary, pp. 16-27.
- Fukuda, T., and Nakagawa, S., (1988). “Dynamic Reconfigurable Robotic System”, *Proc. IEEE Conf. on Robotics and Automation*, vol. 3, pp. 1581-1586.
- Gertler, J. J., (1988). “Survey of model-based failure detection and isolation in complex plants,” *IEEE Control Systems*, vol. 8, no. 6, pp. 3-11.
- Gertler, J. J., (1998). “Fault Detection and Diagnosis in Engineering Systems,” Marcel Dekker, New York, NY.
- Godler, I., and Hashimoto, M., (1998), “Torque control of harmonic drive gears with built-in sensing,” *Proceedings of Annual Conf. IEEE Industrial Electronics Society, IECON*, vol.3, pp.1818-23.



- Hammouri, H., Kinnaert, M., and El Yaagoubi, E. H., (1999). "Observer based approach to fault detection and isolation for nonlinear systems," *IEEE Trans. Automatic Control*, vol. 44, no. 10, pp. 1879-1884.
- Hashimoto, M., (1989). "Robot motion control based on joint torque sensing," *IEEE int. Conf. Robotics and Automation*, pp. 256-261.
- Hashimoto, M., Kiyosawa, Y., and Paul, R. P., (1993). "A torque sensing technique for robots with harmonic drives," *IEEE Transactions on Robotics and Automation* no. 9, pp. 108-116.
- Hui, R., Kircanski, N., Goldenberg, A. A., Zhou, C., Kuzan, P., Wiercienski, J., Gershon, D., and Sinha, P., (1993). "Design of the IRIS facility – a modular, reconfigurable and expandable robot test bed," *Proc. IEEE International Conference on Robotics and Automation*, Vol.3, pp. 155-60.
- Imura, J., Sugie, T., Yokokohji, Y., and Yoshikawa, T., (1991). "Robust control of robot manipulators based on joint torque sensor information," *Proceedings of IEEE/RSJ International Workshop on Intelligent Robots and Systems, Osaka, Japan*, pp. 344-49.
- Ioannou, P. A., and Sun, J., (1996). *Robust adaptive control*. New Jersey: Prentice Hall.
- Isermann, R., (1984). "Process fault detection based on modeling and estimation methods: A survey," *Automatica*, vol. 20, pp. 387-404.
- Isermann, R., (1993). "Fault diagnosis of machines via parameter estimation and knowledge processing," *Automatica*, vol. 29, pp. 815-835.
- Jiang, B., Wang, J. L., and Soh, Y. C., (2002). "An adaptive technique for robust diagnosis of faults with independent effect on system outputs," *Int. Journal of Control*, vol. 75, pp. 792-802.
- Jiang, B., Staroswiecki, M., and Cocquempot, V., (2004). "Fault estimation in nonlinear uncertain systems using robust/sliding mode observers," *IEE Proc. Control Theory. Appl.*, vol. 151, pp. 29-37.
- Kosuge, K., Takeuchi, H., and Furuta, K., (1990). "Motion control of a robot arm using

- joint torque sensors,” *IEEE Trans. on Robotics and Automation*, Vol. 6, No. 2, pp. 258-63.
- Liu, G., and Goldenberg, A. A., (1996a). “Uncertainty decomposition-based robust control of robot manipulators,” *IEEE Transactions on Control Systems Technology*, Vol. 4, No. 4, pp. 384-93.
- Liu, G., and Goldenberg, A. A., (1996b). “Comparative study of robust saturation control of robot manipulators: analysis and experiments,” *International Journal of Robotics Research*, Vol. 15, No. 5, pp. 473-91.
- Liu, G., and Goldenberg, A. A., (1997). “Robust control of robot manipulators based on dynamics decomposition,” *IEEE Transactions on Robotics and Automation*, Vol. 13, No. 5, pp. 783-89.
- Liu, G., (2001). “Control of robot manipulators with consideration of actuator performance degradation and failures,” *Proc. IEEE International Conf. Robotics and Automation*, pp. 2566-2571.
- Liu, G., (2002). “Decomposition-based friction compensation of mechanical systems,” *Mechatronics*, Vol. 12, No. 5, pp. 755-69.
- Liu, G., Goldenberg A. A., and Zhang, Y., (2004). “Precise slow motion control of a direct-drive robot arm with velocity estimation and friction compensation,” *Mechatronics*, vol. 14, no. 7, pp. 821-834.
- Liu, G., Abdul, S., and Goldenberg, A. A., (2006). “Stabilizing modular and reconfigurable robot joint by joint using torque sensing,” *IEEE International Conference on Mechatronics and Automation*, pp. 384-389, Luoyang, China.
- Liu, G., Abdul, S., and Goldenberg, A. A., (2008). “Distributed control of modular and reconfigurable robot with torque sensing,” *Robotica*, vol. 26, no. 1, pp. 75-84.
- Luh, J. Y. S., Walker, M. W., and Paul, R. P., (1980). “Resolved acceleration control of mechanical manipulator,” *IEEE Trans. Automatic Control*, vol. AC-25, pp. 258–263.
- Maciejewski, A. A., (1990). “Fault tolerant properties of kinematically redundant

- manipulators,” *Proc. IEEE Conference of Robotics and Automation*, vol. 1, pp. 638-42.
- Matsumaru, T., (1995). “Design and control of the modular robot system: TOMMS”, *IEEE Proc. of Int. conf. on Robotics and Automation*, pp. 2125-2131.
- McIntyre, M. L., Dixon, W. E., Dawson, D. M., and Walker, I. D., (2005). “Fault identification for robot manipulators,” *IEEE Trans. on Robotics*, vol. 21, no.5, pp. 1028-1034.
- Melek, W. W., and Goldenberg, A. A., (2003). “Neurofuzzy control of modular and reconfigurable robots,” *IEEE/ASME Transactions on Mechatronics*, Vol. 8, No. 3, pp.381-89.
- Melek, W. W., and Najjaran, H., (2005). “Study of the effect of external disturbances on the position control of IRIS modular and reconfigurable manipulator,” *IEEE Int. Conf. on Mechatronics and Automation*, pp. 144-47.
- Middleton, R., and Goodwin, G. C., (1988). “Adaptive computed torque control for rigid link manipulators,” *Systems & Control Lett.*, vol. 10, pp. 9–16.
- Park, D. I., Lee, S. H. and Kim, S. H., (2003). “Torque distribution using a weighted pseudoinverse in a redundantly actuated mechanism,” *Advanced Robotics*, vol. 17, no. 8, pp. 807-820.
- Patton, R. J., Frank, P. M., and Clark, R. N., editors, (1989). *Fault Diagnosis in Dynamic Systems: Theory and Applications*, pp. 477-508. Prentice Hall, Englewood Cliffs, NJ.
- Setchi, R. M., and Lagos, N., (2004). “Reconfigurability and reconfigurable manufacturing systems: state-of-the-art review,” *Proc. IEEE Int. Conf. Industrial Informatics*, pp. 529-35.
- Shin, J. H., and Lee, J-J., (1999). “Fault detection and robust fault recovery control for robot manipulators with actuator failures,” *Proc. IEEE Int. Conf. on Robotics and Automation*, Detroit, MI, pp. 861-866.
- Slotine, J. J. E., and Li, W., (1987). “On the adaptive control of robot manipulators,” *International Journal of Robotics Research*, vol. 6, pp. 49–59.
- Slotine, J. J. E., and Li, W., (1991). “*Applied nonlinear control*,” Prentice-Hall.

- Spong, M. W., and Vidyasagar, M., (1989). *Robot dynamics and control*. New York: Wiley.
- Sreevijayan, D., Tosunoglu, S., and Tesar, D., (1994). "Architectures for fault tolerant mechanical systems," *Proc. IEEE Mediterranean Electrotechnical Conference*, Antalya, Turkey, pp. 1029-1033.
- Staroswiecki, M., and Comtet-Varga, G., (2001). "Analytical redundancy relations for fault detection and isolation in algebraic dynamic systems," *Automatica*, vol. 37, pp. 687-699.
- Stokic, D., and Vukobratovic, M., (1993). "Historical perspectives and state of the art in joint force sensory feedback control of manipulation robots," *Robotica*, no. 11, pp. 149-157.
- Taghirad, H., (1997). "On robust H-infinity torque control of harmonic drive systems," Ph.D Thesis, McGill University.
- Taghirad, H., and Belanger, P.R., (1998). "Torque ripple and misalignment torque compensation for the built-in torque sensor of harmonic drive systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 47, no. 1, pp. 309-316.
- Tesar, D., Sreevijayan, D., and Price, C., (1990). "Four level fault tolerance for manipulator design for space operations," *First International Symposium on Measurement and Control in Robotics*, Houston, Texas.
- Ting, Y., Tosunoglu, S., and Tesar, D., (1993). "A control structure for fault tolerant operation of robotic manipulators," *Proc. IEEE Int. Conf. on Robotics and Automation*, Atlanta, GA, pp. 684-690.
- Ting, Y., Tosunoglu, S., and Freeman, R., (1994). "Torque redistribution method for fault recovery in redundant serial manipulators," *Proc. International Conference of Robotics and Automation*, pp. 1396-1401.
- Tomita, K., Murata, S., Yoshida, E., Kurokawa, H., and Kokaji, S., (1996). "Reconfiguration method for a distributed mechanical system," *Distributed Autonomous Robotic Systems*, Vol. 2, pp. 17-25.
- Vemuri, A. T., and Polycarpou, M. M., (1997). "Robust nonlinear fault diagnosis in

- input-output systems,” *Int. Journal of Control*, vol. 68, pp. 343-360.
- Vemuri, A. T., (2001). “Sensor bias fault diagnosis in a class of nonlinear systems,” *IEEE Trans. on Automatic Control*, vol. 46, no. 6, pp. 949-954.
- Vischer, D., and Khatib, O., (1995). “Design and development of high-performance torque-controlled joints,” *IEEE Transactions on Robotics and Automation*, no. 11, pp. 537-544.
- Visinsky, M. L., Walker, I. D., and Cavallaro, J. R., (1993). “Layered dynamic fault detection and tolerance for robots,” *IEEE Int. Conf. Robotics and Automation*, pp. 180-187.
- Visinsky, M. L., Cavallaro, J. R., and Walker, I. D., (1995). “A dynamic fault tolerance framework for remote robots,” *IEEE Trans. on Robotics and Automation*, vol. 11, no. 4, pp. 477-490.
- Walker, B. K., (1989). “Fault detection threshold determination using Markov theory,” In R. J. Patton, P. M. Frank and R. N. Clark, editors, *Fault Diagnosis in Dynamic Systems: Theory and Applications*, pp. 477-508. Prentice Hall, Englewood Cliffs, NJ.
- Walker, B. K., and Gait, E., (1979). “Fault detection threshold determination techniques using Markov theory,” *Journal of Guidance, Control and Dynamics*, vol. 2, no.4, pp. 313-319.
- Wu, E. C., Hwang, J. C., and Chaldek, J. T., (1993). “Fault tolerant joint development for the space shuttle remote manipulator system: analysis and development,” *IEEE Trans. on Robotics and Automation*, vol. 9, no.5, pp. 675-684.
- Xi, F., Xu, Y., and Xiong, G., (2006). “Design and analysis of a re-configurable parallel robot,” *Mechanism and Machine Theory*, vol. 41, pp. 191-211.
- Xian, B., Dawson, D. M., de Queiroz, M. S., and Chen, J., (2004). “A continuous asymptotic control strategy for uncertain nonlinear systems,” *IEEE Trans. Automatic Control*, vol. 49, no.7, pp. 1206-1211.
- Yim, M., Shen, W.-M., Salemi, B., Rus, D., Moll, M., Lipson, H., Klavins, E., and

- Chirikjian, G. S., (2007). "Modular self-reconfigurable robot systems - challenges and opportunities for future," *IEEE Robotics and Automation Magazine*, Vol. 14, No. 1, pp. 43-52.
- Yim, M., Duff, D. G., and Roufas, K. D., (2000). "Polybot: a modular reconfigurable robot," *Proc. IEEE Int. Conf. Robotics and Automation*, Vol. 1, pp. 514-20.
- Yim, M., (1994). "*Locomotion with a Unit-modular Re-configurable Robot*", Ph.D. Thesis, Stanford University.
- Zhang, G., and Furusho, J., (1998). "Control of robot arms using joint torque sensors," *IEEE Control Systems*, vol. 18, no. 1, pp. 48-54.
- Zhang, Q., Basseville, M., and Benveniste, A., (1998). "Fault detection and isolation in non-linear dynamic systems: a combined input-output and local approach," *Automatica*, vol. 34, pp. 1359-1373.
- Zhang, X., Polycarpou, M., and Parsini, T., (2002). "A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems," *IEEE Trans. on Automatic Control*, vol. 47, no. 4, pp. 576-593.
- Zhu, W.-H., and Lamarche, T., (2007). "Modular robot manipulators based on virtual decomposition control," *Proc. IEEE Int. Conf. Robotics and Automation*, Roma, Italy, pp. 2235-40.



## Appendix A

The dynamics of each link is divided into two dynamic systems, namely the motor system and the link system. Then the following coordinate are defined: the origin of the coordinate frame  $\Sigma_i$  of the  $i^{th}$  link is set on the  $i^{th}$  joint axis. The Z axis of  $\Sigma_i$  is selected such that it aligns with the  $i^{th}$  joint axis, and the unit vector in the direction of the Z axis of  $\Sigma_i$  is denoted by  $z_i$ . The  $i^{th}$  motor which drives the  $i^{th}$  joint is fixed to the  $(i-1)^{th}$  link, and the origin of coordinate frame  $\Sigma_{mi}$  of the  $i^{th}$  motor is set on the axis of rotation of the rotor in the  $i^{th}$  motor, called the  $i^{th}$  rotor. The Z axis of  $\Sigma_{mi}$  is selected in such a way that it aligns with the axis of rotation of the  $i^{th}$  rotor, and the unit vector in the direction of the Z axis of  $\Sigma_{mi}$  is denoted by  $z_{mi}$ . Let the moment of inertia of the  $i^{th}$  rotor about the axis of rotation be  $I_{mi}$ . The output torque of the  $i^{th}$  rotor is denoted by  $\tau_i$  and  $\tau_{ji}$  denoted the coupling forces by the other motor systems and link systems.  $n_{mi}$  denotes the moment vector exerted on the  $i^{th}$  rotor by the  $(i-1)^{th}$  link.  $n_{si}$  denotes the moment vector exerted on the  $i^{th}$  through the torque sensor by the  $i^{th}$  link, which is expressed in the reference frame.  $\tilde{I}_{mi}$ , expressed in reference frame, is the inertia tensor of the  $i^{th}$  rotor.  $\omega_i$  and  $\omega_{mi}$ , expressed in reference frame, represent the angular velocity vectors of the  $i^{th}$  link and  $i^{th}$  rotor respectively.  $q_{mi}$  is an angle about the axis of rotation of the  $i^{th}$  rotor.

Then from Euler's equation,

$$n_{mi} = \tilde{I}_{mi} \dot{\omega}_{mi} + \omega_{mi} \times (\tilde{I}_{mi} \omega_{mi}) + n_{si} \quad (A.1)$$

Using the assumption A2, in Chapter 2,



$$\begin{aligned}
\dot{q}_{mi} &= \gamma_i \dot{q}_i \\
\omega_{mi} &= \omega_{i-1} + \dot{q}_{mi} z_{mi} \\
\dot{\omega}_{mi} &= \dot{\omega}_{i-1} + \gamma_i \ddot{q}_i z_{mi} + \omega_{i-1} \times (\gamma_i \dot{q}_i z_{mi}) \\
n_{si} &= \tau_{Ji} z_{mi} / \gamma_i
\end{aligned} \tag{A.2}$$

Substituting these relations into (A.1), the following equations are obtained:

$$\begin{aligned}
\tau_i &= z_{mi}^T n_{mi} \\
&= z_{mi}^T \tilde{I}_{mi} \dot{\omega}_{i-1} + \left( z_{mi}^T \tilde{I}_{mi} z_{mi} \right) \gamma_i \ddot{q}_i \\
&\quad + z_{mi}^T \tilde{I}_{mi} \left( \omega_{i-1} \times \gamma_i \dot{q}_i z_{mi} \right) + z_{mi}^T \left[ \omega_{mi} \times \left( \tilde{I}_{mi} \omega_{mi} \right) \right] \\
&\quad + \tau_{Ji} / \gamma_i
\end{aligned} \tag{A.3}$$

Using the assumption A1, it is seen that the third and fourth terms of the right hand of (A.3) is equal to zero. And since

$$\dot{\omega}_{i-1} \triangleq \begin{cases} 0 & \text{if } i=1 \\ z_1 \ddot{q}_1 & \text{if } i=2 \\ \sum_{j=1}^{i-1} z_j \ddot{q}_j + \sum_{j=2}^{i-1} \sum_{k=1}^{j-1} (z_k \times z_j) \dot{q}_k \dot{q}_j & \text{if } i \geq 3 \end{cases} \tag{A.4}$$

And  $z_{mi}^T \tilde{I}_{mi} = I_{mi} z_{mi}^T$ , therefore from (A.3),

$$\tau_i = I_{mi} \gamma_i \ddot{q}_i + I_{mi} \sum_{j=1}^{i-1} z_{mi}^T z_j \ddot{q}_j + I_{mi} \sum_{j=2}^{i-1} \sum_{k=1}^{j-1} z_{mi}^T (z_k \times z_j) \dot{q}_k \dot{q}_j + \tau_{Ji} / \gamma_i \tag{A.5}$$

where the second and third terms is equal to zero when  $i=1$ , and the third term is equal to zero when  $i=2$ .

## Appendix B

### B.1 Dynamic Model of 3-DOF planar manipulator

The dynamic equations of motion for a three joint modular robot operating in a horizontal plane are given by

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + f(q, \dot{q}) = \tau$$

where  $D(q)$  is the  $3 \times 3$  inertia matrix,  $C(q, \dot{q})\dot{q}$  is the  $3 \times 1$  vector of centrifugal and Coriolis torques,  $f(q, \dot{q})$  is the  $3 \times 1$  vector of frictional torques and  $\tau$  is the  $3 \times 1$  vector of joint torques. The matrices are given as

$$D = \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix} \quad C = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \quad \tau = [\tau_1 \quad \tau_2 \quad \tau_3]^T$$

$$F(q, \dot{q}) = [f_1(q_1, \dot{q}_1) \quad f_2(q_2, \dot{q}_2) \quad f_3(q_3, \dot{q}_3)]^T$$

where,

$$D_{13} = D_{31} = I_3 + m_3 l_{c3}^2 + m_3 l_2 l_{c3} c_3 + m_3 l_1 l_{c3} c_{23},$$

$$D_{11} = I_1 + I_2 + I_3 + m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + m_3 l_1^2 + m_3 l_2^2 + m_3 l_{c3}^2 \\ + 2m_2 l_1 l_{c2} c_2 + 2m_3 l_1 l_2 c_2 + 2m_3 l_2 l_{c3} c_3 + 2m_3 l_1 l_{c3} c_{23}$$

$$D_{21} = D_{12} = I_2 + I_3 + m_2 l_{c2}^2 + m_3 l_2^2 + m_3 l_{c3}^2 + m_2 l_1 l_{c2} c_2 \\ + m_3 l_1 l_2 c_2 + 2m_3 l_2 l_{c3} c_3 + m_3 l_1 l_{c3} c_{23}$$

$$D_{22} = I_2 + I_3 + m_2 l_{c2}^2 + m_3 l_2^2 + m_3 l_{c3}^2 + 2m_3 l_2 l_{c3} c_3$$

$$D_{23} = D_{32} = I_3 + m_3 l_{c3}^2 + m_3 l_2 l_{c3} c_3$$

$$D_{33} = I_3 + m_3 l_{c3}^2$$

and  $c_2 = \cos(q_2)$ ,  $c_3 = \cos(q_3)$ ,  $c_{23} = \cos(q_2 + q_3)$ ,  $s_2 = \sin(q_2)$ ,  $s_{23} = \sin(q_2 + q_3)$ ,  $m_i$  is the mass of  $i^{th}$  link,  $l_i$  is the length of  $i^{th}$  link,  $l_{ci}$  is the distance from joint to the centre of mass of  $i^{th}$  link,  $I_i$  is the inertia of  $i^{th}$  link. The link masses and inertias include both

the actuator masses and actuator inertias. The joint friction for the  $i^{th}$  joint is modeled as

$$f_i(q_i, \dot{q}_i) = (f_{ci} + f_{si} \exp(-f_{\tau i} \dot{q}_i^2)) \text{sgn}(\dot{q}_i) + b_i \dot{q}_i + f_{qi}(q_i, \dot{q}_i)$$

The Coriolis matrix elements are given by

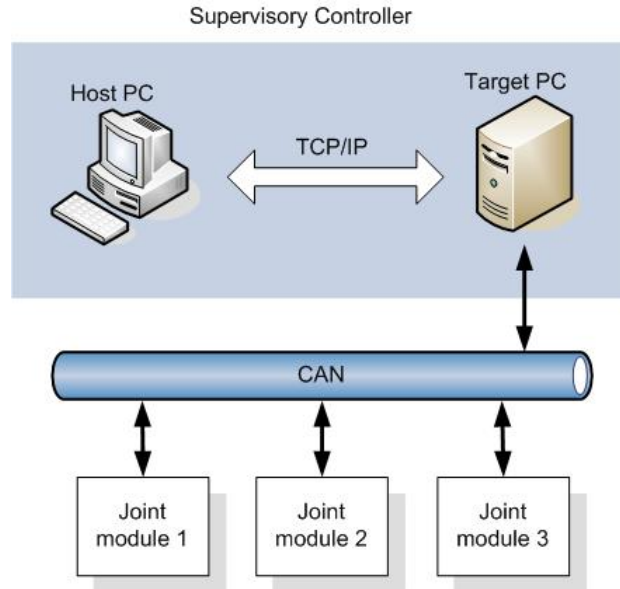
$$\begin{aligned} C_{11} &= c_{111}\dot{q}_1 + c_{112}\dot{q}_2 + c_{113}\dot{q}_3, C_{12} = c_{121}\dot{q}_1 + c_{122}\dot{q}_2 + c_{123}\dot{q}_3 \\ C_{13} &= c_{131}\dot{q}_1 + c_{132}\dot{q}_2 + c_{133}\dot{q}_3, C_{21} = c_{211}\dot{q}_1 + c_{212}\dot{q}_2 + c_{213}\dot{q}_3 \\ C_{22} &= c_{221}\dot{q}_1 + c_{222}\dot{q}_2 + c_{223}\dot{q}_3, C_{23} = c_{231}\dot{q}_1 + c_{232}\dot{q}_2 + c_{233}\dot{q}_3 \\ C_{31} &= c_{311}\dot{q}_1 + c_{312}\dot{q}_2 + c_{313}\dot{q}_3, C_{32} = c_{321}\dot{q}_1 + c_{322}\dot{q}_2 + c_{323}\dot{q}_3 \\ C_{33} &= c_{331}\dot{q}_1 + c_{332}\dot{q}_2 + c_{333}\dot{q}_3 \end{aligned}$$

where

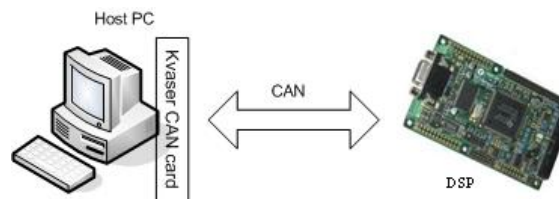
$$\begin{aligned} c_{133} &= -m_3 l_2 l_{c3} s_3 - m_3 l_1 l_{c3} s_{23}, \quad c_{312} = m_3 l_2 l_{c3} s_3, \\ c_{211} &= m_3 l_1 l_2 s_2 + m_2 l_1 l_{c2} s_2 + m_3 l_1 l_{c3} s_{23}, \quad c_{213} = -m_3 l_2 l_{c3} s_3, \\ c_{311} &= m_3 l_2 l_{c3} s_3 + m_3 l_1 l_{c3} s_{23}, \quad c_{321} = c_{312}, \quad c_{131} = c_{113}, \\ c_{233} &= -m_3 l_2 l_{c3} s_3, \quad c_{322} = m_3 l_2 l_{c3} s_3, \quad c_{223} = -m_3 l_2 l_{c3} s_3, \\ c_{112} &= -m_2 l_1 l_{c2} s_2 - m_3 l_1 l_2 s_2 - m_3 l_1 l_{c3} s_{23}, \\ c_{113} &= -m_3 l_2 l_{c3} s_3 - m_3 l_1 l_{c3} s_{23}, \quad c_{123} = -m_3 l_2 l_{c3} s_3 - m_3 l_1 l_{c3} s_{23}, \\ c_{122} &= -m_2 l_1 l_{c2} s_2 - m_3 l_1 l_2 s_2 - m_3 l_1 l_{c3} s_{23}, \\ c_{132} &= c_{123}, \quad c_{232} = c_{223}, \\ c_{231} &= c_{213}, \quad c_{121} = c_{112}, \\ c_{111} &= c_{212} = c_{221} = c_{222} = 0, \\ c_{331} &= c_{313} = c_{323} = c_{332} = c_{333} = 0 \end{aligned}$$

## Appendix C

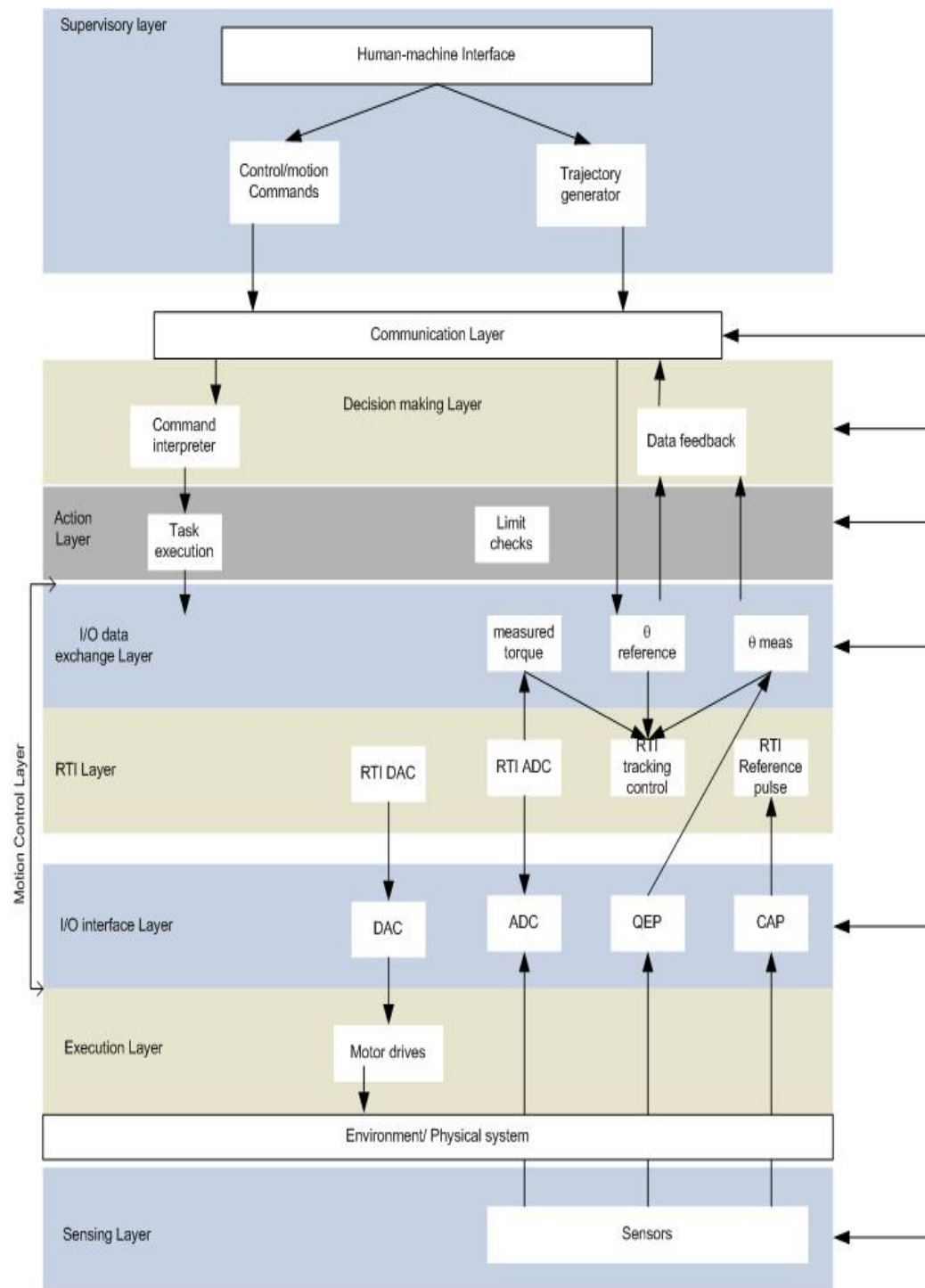
### C.1 Controller Architecture and Flow Diagrams



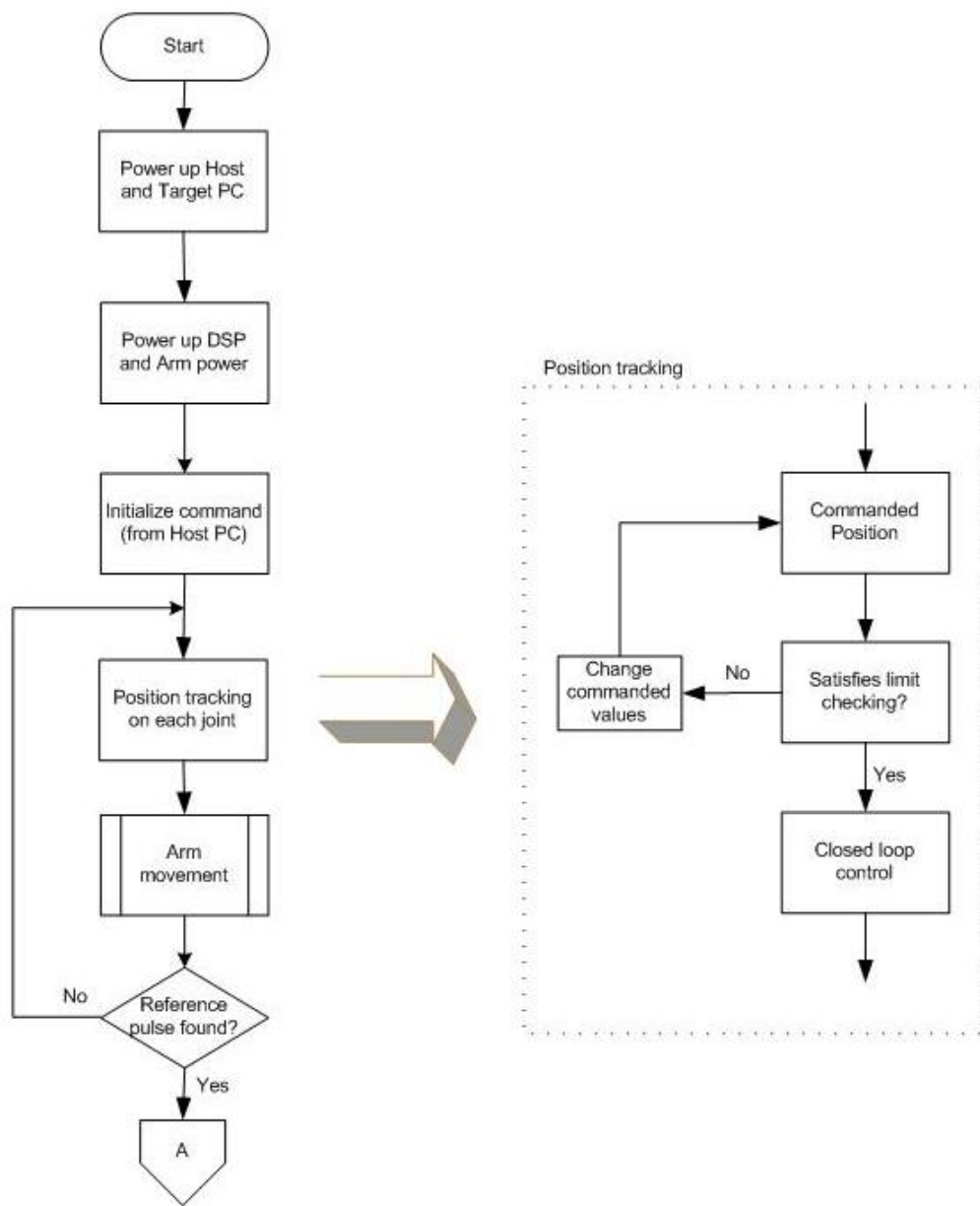
**Figure C-1. Control system architecture of modular and reconfigurable robot**



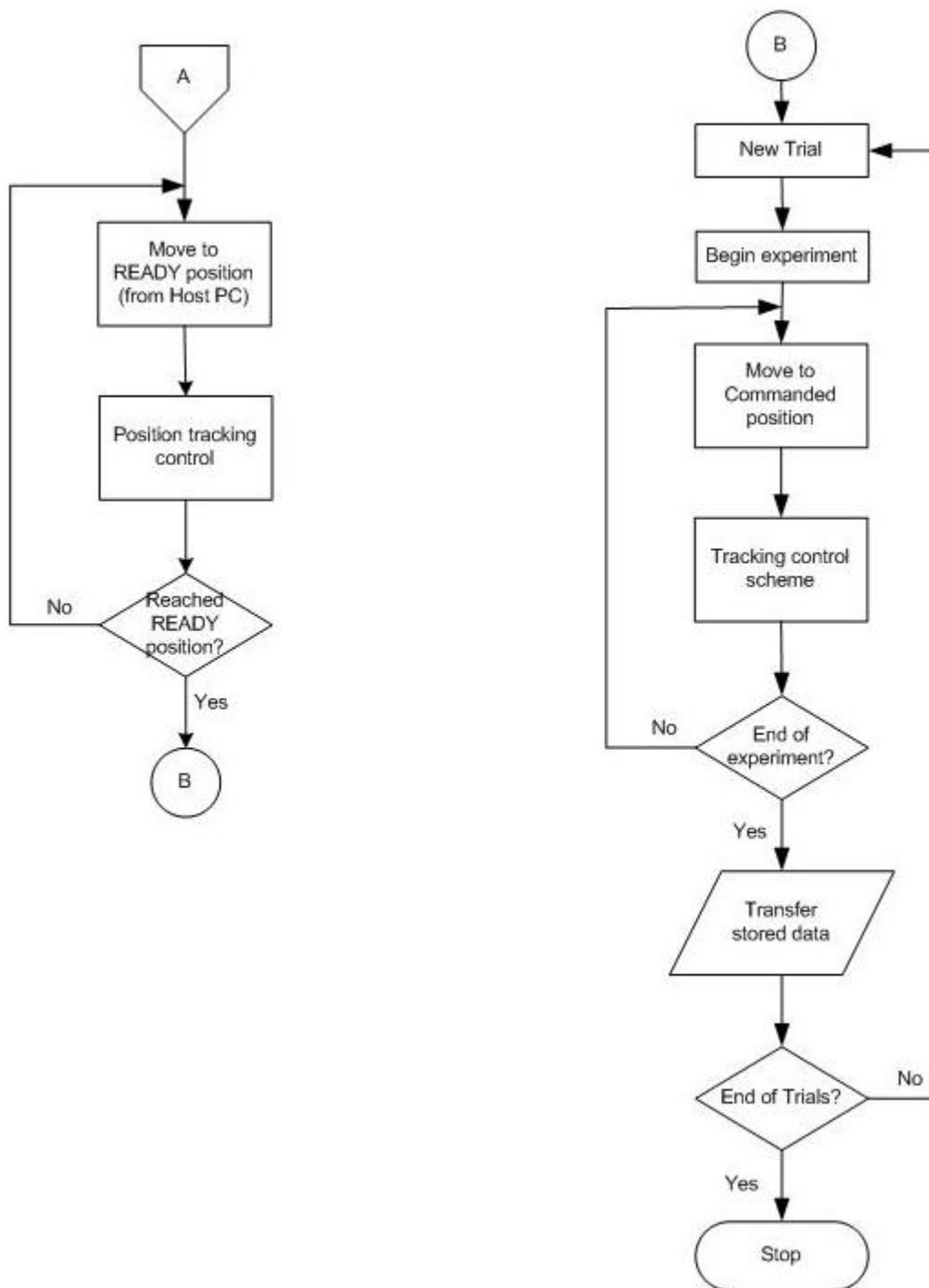
**Figure C-2. Host PC - DSP controller communication over CAN**



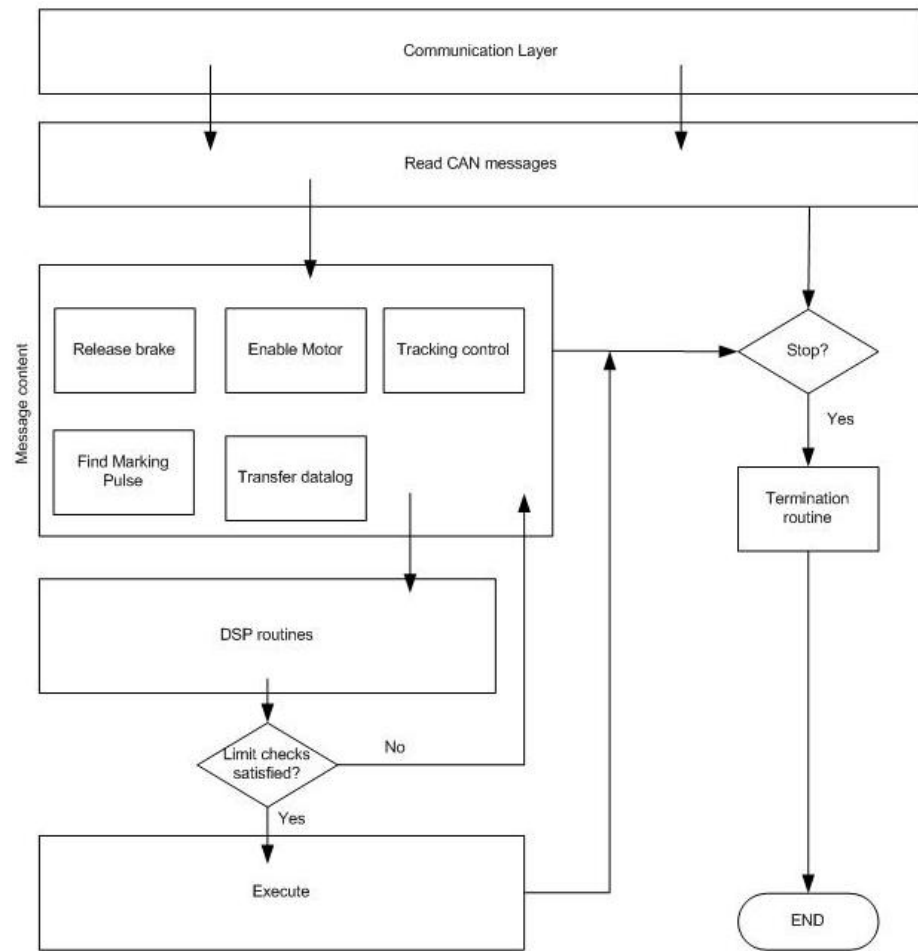
**Figure C-3. Architecture of MRR control system**



**Figure C-4. Flow chart for the initialization of MRR module (Level 1)**

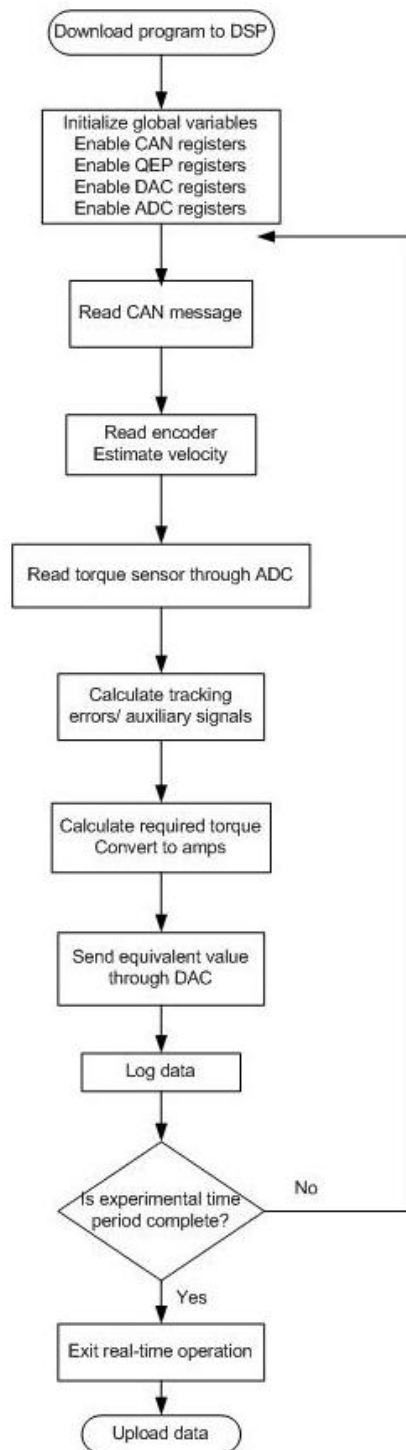


**Figure C-5. Flow chart of the MRR module operation (Level 2)**

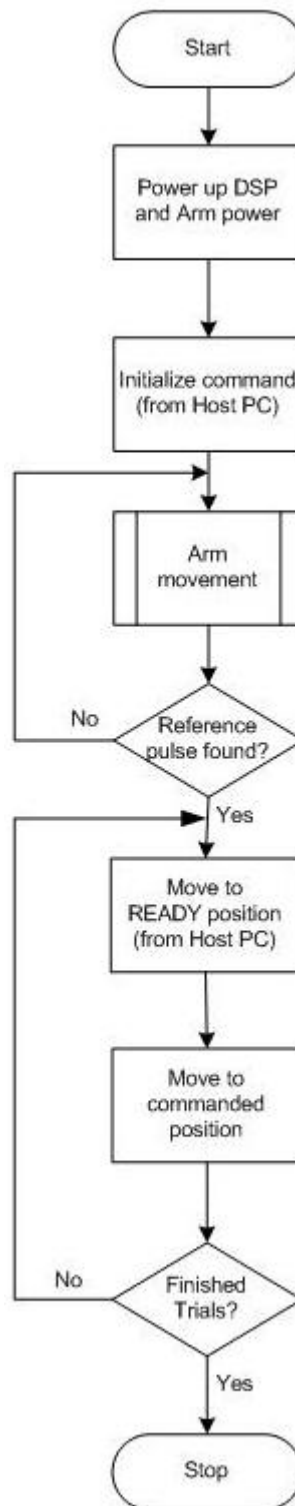


**Figure C-6. Message interpreter and execution at DSP from communication layer**





**Figure C-7. Flow chart of sequence of tasks on DSP controller, at each sample period**



**Figure C-8. Flow chart for the operation of 3DOF MRR system.**