FORCE-CONTROLLED TRANSCRANIAL MAGNETIC STIMULATION (TMS) ROBOTIC SYSTEM

WAN NURSHAZWANI WAN ZAKARIA

The thesis is submitted in fulfillment of the requirements for the Degree of Doctor of Philosophy

School of Mechanical and Systems Engineering

Newcastle University

July 2012

This thesis is dedicated to my parents.

ABSTRACT

The use of robots to assist neurologists in Transcranial Magnetic Stimulation (TMS) has the potential to improve the long term outcome of brain stimulation. Although extensive research has been carried out on TMS robotic system, no single study exists which adequately take into account the control of interaction of contact force between the robot and subject's head. Thus, the introduction of force feedback control is considered as a desirable feature, and is particularly important when using an autonomous robot manipulator.

In this study, a force-controlled TMS robotic system has been developed, which consists of a 6 degree of freedom (DOF) articulated robot arm, a force/torque sensor system to measure contact force and real-time PC based control system. A variant of the external force control scheme was successfully implemented to carry out the simultaneous force and position control in real-time. A number of engineering challenges are addressed to develop a viable system for TMS application; simultaneous real-time force and position tracking on subject's head, unknown/varies environment stiffness and motion compensation to counter the force-controlled instability problems, and safe automated robotic system.

Simulation of a single axis force-controlled robotic system has been carried out, which includes a task of maintaining contact on simulated subject's head. The results provide a good agreement with parallel experimental tests, which leads to further improvement to the robot force control. An Adaptive Neuro-Fuzzy Force Controller has been developed to provide stable and robust force control on unknown environment stiffness and motion. The potential of the proposed method has been further illustrated and verified through a comprehensive series of experiments.

This work also lays important foundations for long term related research, particularly in the development of real-time medical robotic system and new techniques of force control mainly for human-robot interaction.

KEY WORDS: Transcranial Magnetic Stimulation, Robotic System, Real-time System, External Force Control Scheme, Adaptive Neuro-Fuzzy Force Controller

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my supervisor Dr. Robert Bicker, whose advice, stimulating suggestions and encouragement helped me in all the time of research and writing of this thesis. I also want to thank my second supervisor, Prof. Stuart Baker for his support and direction of this work.

I would like to acknowledge my colleague Xiang Yi, who has helped me in developing the robotic system. I would like to thank my fellow colleagues in Robotics and MEMS group, who shared the joys and travails during this research. I give my thanks to technical staff of the School of Mechanical and System Engineering, Newcastle University for their valuable assistance with this work.

I am truly indebted to the Ministry of Higher Education (MOHE) of Malaysia and Universiti Tun Hussein Onn (UTHM) for their scholarship award which allows me to commence this thesis in first instance.

I am especially grateful for my husband Nik Farhan for his continuous support and encouragement, and the understanding shown over a few years although being apart throughout this work. Finally and most importantly, my sincere thanks go to my parents for their love, support and encouragement.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background and motivation of the research

Recent developments in the neuroscience area have heightened the need for transcranial magnetic stimulation (TMS) as a non-invasive and powerful method with which to study human brain behaviour. Originally TMS was used in clinical neurology to study motor cortex conduction time and to measure synapses in the brain and spinal cord. Currently, it is being widely used as a research tool to study aspects of human brain physiology including motor function, vision, language and the pathophysiology of brain disorders. TMS has also become an important technique for changing the activity of brain neurons and the functions they sub serve, apart from being a significant adjunct to brain imaging and mapping techniques.

In the TMS procedure, an electromagnetic coil is placed over the head, and a large but brief current pulse is produced and passed through the scalp. The changing magnetic field which this generates induces a current in the conductive tissue of the head including the underlying cerebral cortex. The cortical activity produced by TMS is critically dependent on coil location and orientation. Currently, the coil is positioned manually which can be a problem for precise clinical evaluation and experiments of the TMS technique, where the coil is fixed relative to the head using devices such as face masks, helmets or bite bars which can be uncomfortable for the subject and severely limit experimental duration. In addition, the search for an optimal site to activate the motor cortex has to be carried out by an expert who moves the coil between scalp locations whilst viewing the previous response. Stimulation of other areas is more problematic as no electrophysiological response can be used to indicate correct positioning.

The purpose of this study is to increase the reliability of TMS of the cerebral cortex. This will be achieved by designing and building a robot system which will position and hold the magnetic stimulus coil over a fixed location on the head, at a fixed orientation. The system will be a valuable tool in helping patients who suffer from neurological diseases, such as Parkinson's disease, anxiety disorders, schizophrenia, movement disorders, epilepsy and an alternative treatment for depressions. Transcranial magnetic stimulation is a promising new research field, but more work remains to be done before it can be fully integrated in therapeutic applications in psychiatry and neurology.

1.2 The Research Gaps

Although the TMS technique is very useful, it is not yet widely accepted because of the observed inconsistency of efficiency between subjects. TMS coils produce very dense and defined fields, which mean that a small movement of the coil position leads to a substantial change of the electromagnetic field delivered to a target nerve. It is due to the difficulty to indicate correct positioning with the current available stimulation systems which leads to poor repeatability. This explains the vital need for technologies to support precise navigation and positioning of TMS coils.

Several researchers have developed image-guided TMS to overcome this problem [Herwig et al., 2001; Lancaster et al., 2004; Matthäus et al., 2006a]. By registering MRI data with the patient's head and by tracking the coil, the area of stimulation can be determined easily [Ettinger et al., 1998]. Interleaving TMS and functional brain imaging offers much promise. Matthäus et al. [2006a; 2006b] pointed out a major problem with this kind of technique; exact stimulation is hard to achieve at a pre-defined point, particularly if the site is to be stimulated several times or continuously as in the treatment of depression. Either head movements must be followed by manually adjusting the coil continuously or the head must be fixed. George et al. [1999; 2006b] discusses two problems for combining TMS with functional and structural neuroimaging. One of the limitations with this technique is sometimes it is difficult to match the imaging technique to the temporal duration of TMS. Additional problems have concerned the interference produced by TMS with image acquisition.

A lot of work has been devoted to the development of TMS to assist the neurologist during stimulation sessions. This leads other promising applications of robotics in TMS to position the coil for efficient and reliable TMS procedures. Robot assistance in TMS procedure has been reported by a number of researchers [Matthäus et al., 2006a;

Matthäus et al., 2006b; Finke et al., 2008; Lebosse et al., 2008]. These existing TMS robotic systems, which aim at replacing the hand and the arm of neurologist, are particularly focused on image-guided by means of navigation system to position the coil on subject's head.

Despite its safety and efficacy, current researchers are unable to address the force control as an important feature of the TMS procedure. Most of the research to date has tended to focus on image-guided and navigation system and neglected the study on contact force between the coil and the subject's head which is crucial. Excessive contact force between the coil and subject's head may harm the subject. It is the aim of this study to develop a novel force-controlled robotic system to perform a safe and reliable efficient evaluation of the TMS technique. Apart from providing an indication of TMS system performance, this new robotic system has its emphasis on real-time control, programmability and repeatability which the robot provides to the advantage for TMS. Robotic systems also have much faster response time with reduced computational delay compared to manually TMS procedures, thus allowing faster detection and reaction to the stimulation process.

1.3 Aims and Objectives

This research aims to design and develop a robust and safe force-controlled TMS robotic system to assist neurologists in medical and clinical practice as well as to make stimulation of the cerebral cortex more reliable. This will be achieved by designing and building a robot system which will position and hold the magnetic stimulus coil over a fixed location on the subject's head at a fixed orientation. The robot will track small head movements made by the subject and will maintain a low contact force between the subject's head and coil. One of the major characteristics of this system is to strongly interact with human environment thus many safety requirements are needed, for it to be capable of working in close proximity to a human subject without risk of injury.

The primary aim of this research is to design an appropriate force control scheme and integral safety system to improve the effectiveness of automated TMS robotic system as well as its efficiency and reliability. Both of these systems will be considered with both development in hardware and software, so that it will be capable of working as an integral part of the robotized TMS system.

To achieve the above aim, the following objectives have been set:

- Undertake an in-depth review of the hand-held TMS procedure, medical robotics system, industrial robot and force control schemes.
- Design an intrinsically automated force-controlled Transcranial Magnetic Stimulation (TMS) robotic system.
- Establish a mathematical model of one degree of freedom TMS robotic system to provide an analytical overview of the factors affecting stability of the control system.
- Develop a force/torque sensor data acquisition system to provide a reliable and real-time measurement.
- Design and develop a real-time robotic control system comprising the robot communication, real-time path control, synchronization approach and software process based structure.
- Develop and implement a force control strategy to maintain a light contact force between the coil and subject's head.
- Design and develop an adaptive Neuro-Fuzzy force controller to improved behaviour of the 6 DOF force-controlled TMS robotic system.
- Evaluate the developed force-controlled TMS robotic system via a series of experiments.

This work is conducted in parallel with Xiang Yi [Yi, 2012] who focused on position tracking control. This study represents, for the first time, a detailed investigation and implementation of force control strategy on TMS robotic system with unknown knowledge of the subject's head characteristics, and thereby constitutes an important contribution to the evaluation of the TMS robotic system.

1.4 Hypothesis

An intelligent decision-making method called an adaptive Neuro-Fuzzy control has a possibility to be utilized in TMS robotic system to improve the effectiveness and reliability of TMS procedure in terms of force-controlled approach. The adaptive Neuro-Fuzzy approach has a unique capability of modeling complex nonlinear processes to arbitrary degrees of accuracy, therefore allowing the robotic system to position and hold the stimulating coil at a fixed target position of unconstrained subject’s head.

1.5 Layout of Thesis

Initial work was devoted to the comprehensive review of the TMS procedure, medical robotic system and force control strategies as described in Chapter 2. Prior to the implementation of force control to the TMS robotic system, a TMS procedure analysis was carried out to identify the control functions required to be incorporated in an automated system as discussed in Chapter 3.

From this evolved the detailed design of the TMS robotic system in Chapter 4. A detail explanation of the hardware configuration and software architecture is also discussed. Chapter 5 provides a developed mathematical model of one degree of freedom TMS robotic system. An analytical analysis of the system control performance is conducted and particularly focuses on the factors affecting the control system stability.

Chapter 6 focuses on the implementation of an external force control scheme on the developed 6 DOF TMS robotic system. Initial commissioning test results using a conventional force controller showed the dynamics of the environment to be a significant parameter in the force control stability. This was found problematic in TMS application as subject’s head characteristic is varied from one to another and can be considered as unknown or uncertainties.

As a consequence, an adaptive Neuro-Fuzzy force controller is designed and implemented on the TMS robotic system as discussed in Chapter 7. This is followed by the evaluation of the proposed force-controlled TMS robotic system in Chapter 8. An analysis of test results is also presented. Finally, Chapter 9 presents the conclusions drawn from the study and the recommendations for future work.

CHAPTER 2

LITERATURE REVIEW

This chapter provides an overview of the topics covered throughout the study, begins with an introduction to Transcranial Magnetic Stimulation (TMS), the principle of operation, localization of the coil and requirements for the TMS procedure are briefly described. Secondly, an example of the conventional TMS procedure on motor cortex is discussed together with its drawbacks. The possibility of developing robot assisted systems in TMS is then investigated and relevant existing researches are discussed. Afterwards, the objectives of the proposal to develop force control strategy in TMS procedure are elucidated.

In order to find the most appropriate force control scheme for the TMS robotic system, several force control schemes will be discussed in section 2.6. It is not intended to cover all existing strategies that have been developed, but rather to highlight the aspects and concepts which will be useful for developing a control algorithm for TMS robotic system. A range of force control schemes are classified and compared, followed by a discussion on application of external force control in medical robot systems together with its advantages.

2.1 Introduction to Transcranial Magnetic Stimulation (TMS)

2.1.1 Basic Principle of Magnetic Stimulation

The principles of electromagnetism that underlie TMS were well known more than a century before its introduction by Barker and colleagues 20 years ago [Barker and Jalinous, 1985; Wassermann et al., 2008]. Electrical current can excite cells such as neurons and muscles. The tingling sensation when short-circuiting a low power battery with fingers is a good example of this. Even a strong electrical pulse can be beneficially applied in medicine, for instance, in cardiac defibrillators.

Electromagnetic theory is both elegant and arcane. Magnetic stimulation can “mimic” direct stimulation with electrical current in the tissue. This has beneficial consequences because the magnetic stimulation can reach the tissue without the need for physical contact. Thus, it can be used to examine the brain and treat disorders of the head painlessly and non-invasively. The mechanism of action is similar for both electrical and magnetic stimulation. In order to stimulate neuronal cells, an electric field E must be applied to the tissues; it forces coherent motion of free charges in intra- and extra-cellular spaces [Ruohonen, 2003]. The physical phenomenon behind magnetic stimulation is electromagnetic induction and is governed by a basic law of physics namely, Faraday’s law:

$$\nabla \times E = -\frac{\partial B}{\partial t} \tag{2-1}$$

In the TMS technique, this relation states that an electric field *E*, and thereby electrical current *I*, is induced in tissue by the time-varying magnetic field *B* from the coil. The solution to the above equation (2-1) gives an estimate of the induced *E* and ignores the effects of the conductivity variations between and within the brain, skull and scalp. Magnetic fields encountered in magnetic stimulation travel freely in air and can easily penetrate through tissue. Therefore, magnetic stimulation can readily reach brain cells even through a high resistance skull. Figure 2-1 below shows the basic principle of TMS.

A magnetic field *B* that induces an electric field *E* is generated by current pulse ***I(t)*** in the coil. The lines of *B* go through the coil and the lines of *E* follow the shape of the coil. The upper-right drawing illustrates two pyramidal axons, together with a typical orientation of the intracranial *E*. The *E*-field affects the transmembrane potential which may lead to local membrane depolarization and firing of the neurons. Events following TMS include: (1) coherent activation of neurons; (2) metabolic and hemodynamic changes; and (3) behavioural changes.

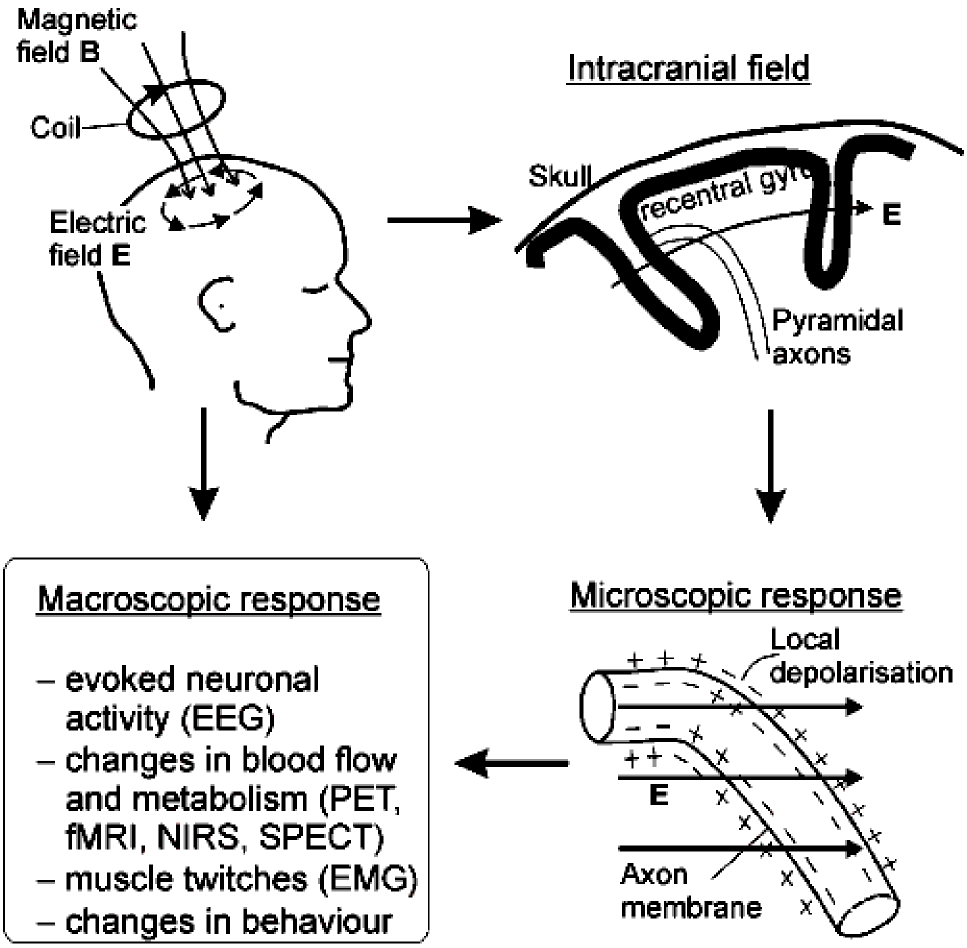


Figure 2-1 Principle of TMS [Ruohonen, 2003]

2.1.2 TMS waveform and current direction

Wassermann et al. [2008] suggest that the effectiveness of TMS in humans is determined by waveform and current direction. Basically there are two pulse configurations of TMS namely single-pulse and paired pulse. For single-pulse TMS, waveform and current direction has been shown to influence motor threshold, motor evoked potential (MEP) latency and silent period duration based on motor cortex and occipital simulations tests [Sommer et al., 2001; Wassermann et al., 2008]. On the other hand, for paired TMS pulses, these factors affect the range of inhibition and facilitation. Nowadays, repetitive TMS (rTMS), capable of delivering a series of high frequency (1-50Hz) pulse is available. This type of pulse produces powerful effects that outlast the period of stimulation. The extent and specificity of the induced motor cortex inhibition and facilitation also depends on TMS waveform and current direction, which enhance clinical efficacy of rTMS. However, rTMS has the potential to cause seizures even in normal individuals. Wassermann et al. [1998] provide safety guidelines describing limits for combinations of frequency, intensity, and train length which should minimize most of the problems.

2.1.3 TMS Coils

The direction of the induced current flow and the size of stimulated area are dependent on the coil shape. There are several available coil types such as circular, figure-of-eight, cap-shaped and conical coils. The field strength from the coil decreases quickly with distance from the coil. The most commonly used type in TMS is the figure-of-eight coil, shown in Figure 2-2(a). This type of coil is more focused and produces strong induced field at the intersection of the two circular coils, compared to a similar sized single coil. Cap-shaped and cone coils are constructed with a pair of circular windings at such an angle with respect to each other that they fit the curvature of the head. These types of coil increase the power at the intersection and are somewhat more effective than the normal planar coils.

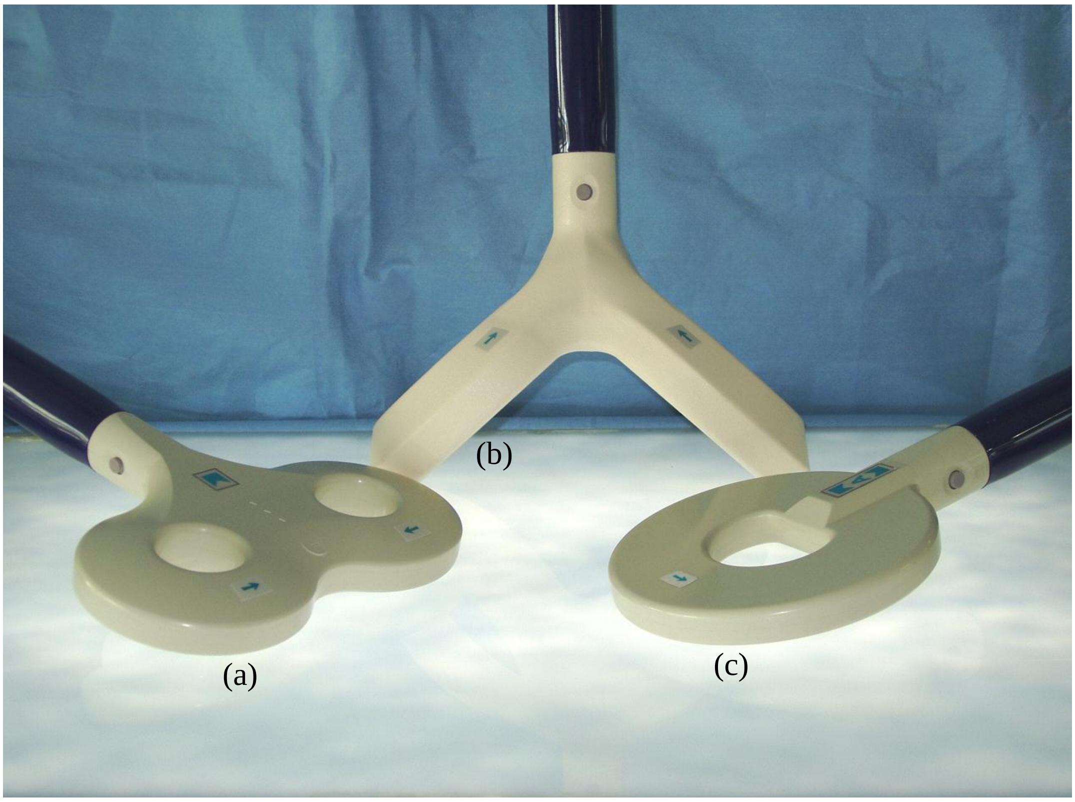


Figure 2-2 TMS coils (a) figure-of-eight, (b) cone-shaped and (c) single shaped coils

2.2 Localization of the TMS Coil

The use of TMS requires knowledge of where on the cortex the magnetic stimulation has to be applied and how it is controlled. Experienced neurologists determine the stimulation site for measuring the latencies of neurons quite easily; the position of the coil over the head is varied until the desired effect is obtained, for example a motor response. This effect defines the optimal stimulation site and this approach is common and practicable. This leads to several localization techniques in the positioning method for TMS, as given in Table 2-1.

Table 2-1 Different method to localize TMS [Herwig et al., 2001]

Principle of localization	Navigational method	Advantage/disadvantage
Functional	Orientation to motor response, visual phenomenon, speech arrest	Verification by function, not applicable if function is not monitorable (e.g. memory)
Anatomical	10-20-system of EEG, defined landmarks	Standardized, not oriented to individual function or cortex anatomy
Functional and anatomical features combined	5cm anterior to motor cortex in prefrontal cortex	Not oriented to individual cortex anatomy, imprecise within cm
Digitizing devices like 'Polhemus Isotrak'	3D-radio frequency localization and co-registration with MRI	Individual localization, resolution approximately 5mm, not applicable during stimulation
Marker in MRI scan, functional imaging	Marker fixed over stimulated area, combination with PET	Individual localization afterwards, not applicable during stimulation, no monitoring
Stereotactical, mechanically guided navigation	3D stereotaxy by joint sensors in mechanical arms	Individual localization, potentially high precision, monitoring, complex mechanical set up
Stereotactical, optically tracked navigation	3D-camera system, detection of light-emitting diodes, co registration with MRI	Individual localization, resolution approximately 2mm, monitoring during stimulation

A multiple positioning method analysis has revealed that the mechanically guided and optically guided navigation of transcranial magnetic stimulation is a practical and comfortable method for stimulating individually selected cortical areas with high precision. However, with these approaches the head has to be fixed strictly to rule out movements during the stimulation, and it is necessary to combine these methods with functional neuroimaging to make it more reliable.

2.3 Medical and Applications of TMS

Nowadays, there are several non-invasive techniques available in neuroscience research and applications such as EEG/MEG and functional imaging. Each method provides a different view of brain function and one in particular may be the best for a specific condition, however, it is advantageous to obtain multiple views for a more complete understanding. Electroencephalography (EEG) and Magnetoencephalography (MEG) are two techniques used for direct measurements of neuronal activity. EEG measures voltage differences produced by the firing of neurons within the brain between different sites along the scalp. The main application of EEG is in the diagnosis of epilepsy, coma and encephalopathy. However, the main disadvantage of this method is scalp potential can be measured only when a sufficient number of cells are active synchronously. In addition, those that are oriented perpendicular to the surface of cortex have more influence compared to those oriented tangentially. MEG is similar to the EGG but measures intracellular currents. In addition, this technique is a better solution to localize neurons because MEG is not distorted by the skull and scalp. However, MEG is blind to radial sources.

Other functional neuroimaging techniques such as Position Emission Tomography (PET) and functional Magnetic Resonance Imaging (fMRI) have good spatial localization but less temporal resolution. PET measures regional cerebral blood flow since synaptic activity increases local metabolism and stimulates changes in perfusion. The fMRI is an indirect technique used to measure neuronal activity in which the oxidation state of hemoglobin in the blood is measured. Blood flow increases more than oxygen extraction with metabolism thus, blood becomes more oxygenated.

Several researchers have used different methods to study a similar physiological process to obtain different points of view, for example in a go/no-go experiment [Hallett, 2007].

The go/no-go experiment is a two-choice reaction time experiment to either move or not move depending on the stimulus. From the experimental results, EEG suggests that something is apparently happening despite a negative fMRI where nothing has appeared. A study using TMS clarifies the confusion. The TMS technique detects inhibition as well as increased excitability in the motor cortex. In a case study by Hallett [2007], the EEG detected what happened throughout its time course whereas neuroimaging did not show any activity, as inhibition is not as demanding a metabolism process as is excitation [Hallett, 2007]. Thus, it can be concluded that TMS can be used in conjunction with the other imaging techniques to improve better understanding particularly in neuroscience research.

TMS studies have also played a useful role in the application of pathophysiology, which is a medical branch used to study functional changes associated with or resulting from diseases. For example, TMS can provide information about different mechanisms for genesis of epileptic seizures and for the modes of action of antiepileptic drugs by assessing cortical excitability. In addition, TMS also can potentially be used to quantify physiological effects in individual patients, and this may be more valuable in some circumstances than anticonvulsant blood levels. Cantello [2002] claims studies with TMS have revealed abnormalities in movement disorders such as Parkinson’s disease, Huntington’s disease and dystonia.

TMS stimulation has been applied to change sypnatic strength to obtain long-lasting influences on the brain, for instance by alterations in dendrite spines. Such logic can be used in therapeutic applications for many neurological disorders such as Parkinson’s diseases, dystonia, stroke and epilepsy. TMS can speed up the reaction time in patients with Parkinson’s disease as well as increasing intracortical inhibition in dystonia. The most extensive use of TMS therapy is for psychiatric conditions, mainly depression in which TMS can deliver equally effective focal therapy more easily and with fewer side effects[Wassermann et al., 2008].

As a conclusion, TMS can be suggested as an excellent non-invasive physiological and therapeutic tool and complements other methods to study physiology, motor and sensory area of human brain. Hallett [2007] suggests TMS can add more power to the clinical neurophysiologist for diagnosis and therapy of neurological disorders.

2.4 Conventional TMS Procedure

The typical TMS procedure is given below as an example. Conventionally, subjects are seated in a comfortable chair with their hands placed on their lap. A TMS coil is placed on the scalp and a brief electrical current is passed through the coil, creating a magnetic pulse that stimulates the outer part of the brain, called the cortex. This may cause twitching of a particular muscle in the hand or arm, depending on the location of the coil relative to the position of the motor cortex, i.e part of the brain to control movement. The procedures may be different depending on the subjects, operators, types of illness and their treatments. The procedure may last for two-three hours for every session. The following section will discuss the clinical procedure requirements to ensure a good outcome of TMS.

2.4.1 Clinical procedure requirement and components

In order to implement TMS technology into a clinical practice and usable configuration, it is essential to consider appropriate and additional features into TMS clinical system to achieve a consistent therapy. In particular, unique and proper procedural aspects of a specific therapy (for example, treating depression) are considered as indicated in Figure 2-3 [Wassermann et al., 2008]. Four fundamental aspects of a TMS stimulator system intended for routine clinical use are listed below:

- Patient positioning and comfort
- Accurate and repeatable coil positioning over the course of many treatment sessions
- Management and analysis of patient data to perceive trends and response
- Incorporation of features to monitor the quality of stimulator operation

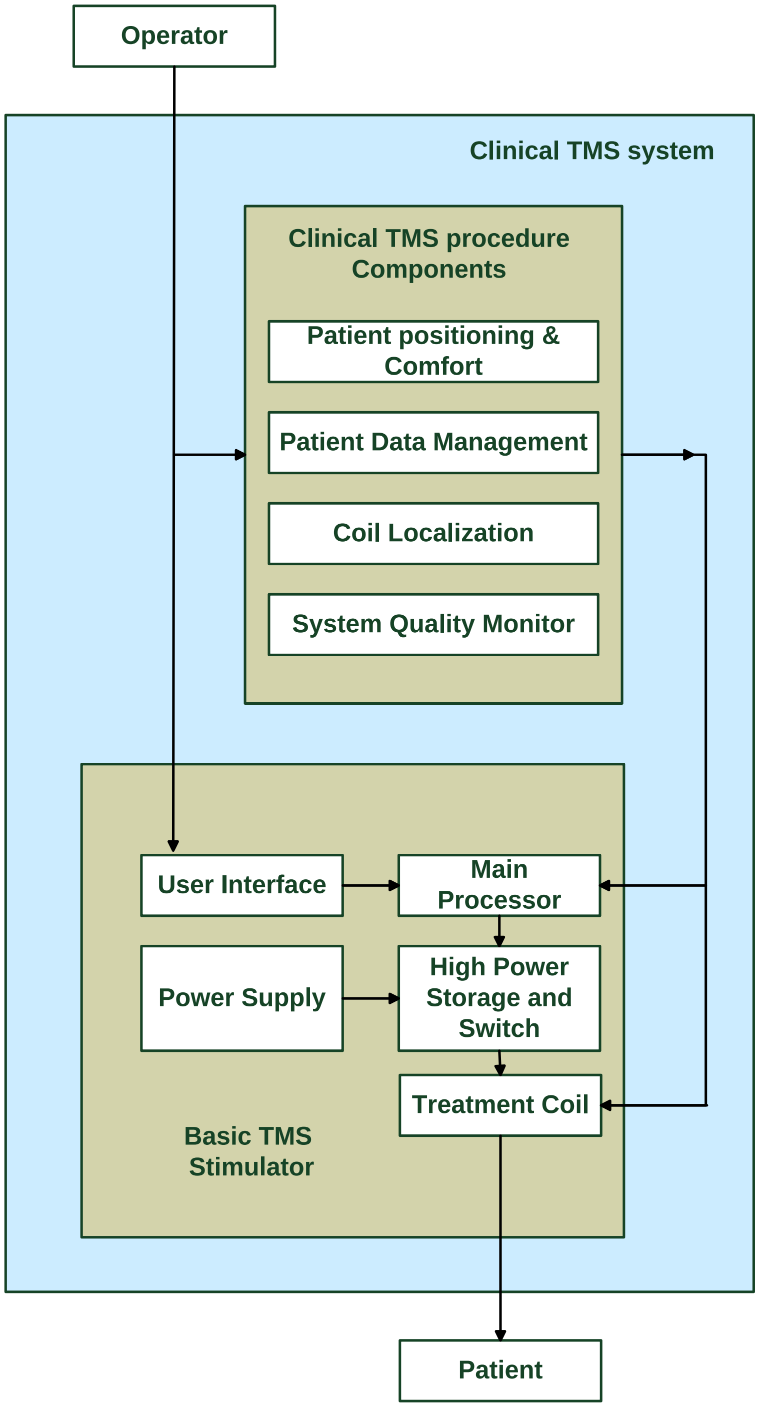


Figure 2-3 Clinical TMS System [Wassermann et al., 2008]

Patient comfort features are important to improve tolerability, to minimize disruptions and undesired movement of the subject relative to the coil as a typical session of treatment may take 10-30 minutes. The clinical system should also provide the clinician with a straightforward means of locating the motor threshold position and navigating from the position to the treatment position [Wassermann et al., 2008]. Several approaches such as markers, swim cap, mechanical alignment and image guided graphical methods have been implemented in order to provide reference points for coil placement. However, all previously mentioned approaches carry various well known limitations that include being uncomfortable for patients, and the three-dimensional image set may limit the clinical practice routine. Optical and ultrasonic coordinate measuring systems offer accurate methods for repeatable coil positioning but cost may prohibit adoption of this kind of technology in typical TMS clinical practice.

It has been suggested that the positioning method and apparatus should be intuitive, repeatable, accurate, comfortable and acceptable to the patients, for example by minimizing restraint and hair disruption. Furthermore, the capability of supporting the coil mass and maintaining its position throughout a prolonged TMS procedure will be necessary without excessively constraining the patient. A contact sensing feature such as force sensor that provides real-time feedback to the operator should be incorporated since the magnetic field diminishes rapidly with distance from the coil. Other data management features must also be incorporated into the TMS clinical system to assist the operator to recall treatment protocols, review patient treatment history, record observations as well as to analyze trends [Wassermann et al., 2008].

2.5 Current research in TMS Robot Assisted Systems

Robot assistance in TMS procedure has been reported by a number of researchers [Matthäus et al., 2006a; Matthäus et al., 2006b; Finke et al., 2008; Lebosse et al., 2008]. These existing TMS robotic systems aimed at replacing the hand and the arm of neurologist are particularly focused on image-guided systems by means of navigation to position the coil on subject head.

2.5.1 A robotic image-guided Transcranial Magnetic Stimulation

Lebosse et al. [2008] propose a new image-guided robotic TMS system that used a seven degree of freedom redundant manipulator arm. The robotic system consists of three subsystems which are a spherical mechanism, an actuated prismatic joint and a spherical serial wrist. The first subsystem deals with the three degree of freedom serial structure (J1, J2 and J3) which is used to provide kinematic positioning. The orientation of the coil around a sphere centered on the subjects head can be achieved using a spherical workspace. J4 which is an actuated prismatic joint is used to control the coil/head including a force sensor that is attached to the coil casing. A hybrid position/force control scheme with a vision-based head position localization system will be implemented. The spherical serial wrists are used to maintain the coil position so it is always tangent to the head surface without changing it centre position. The remaining joint is controlled to follow the cortical column direction. Figure 2-4 presents the kinematics scheme and the CAD model of the robot.

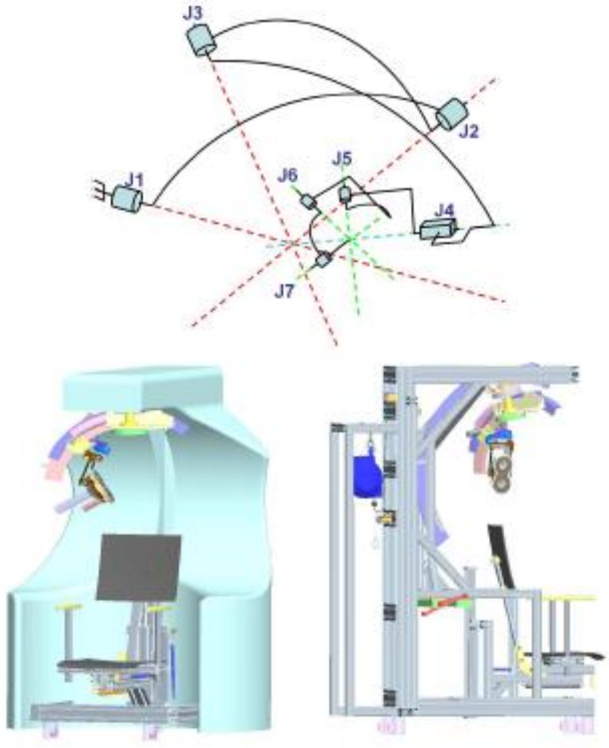


Figure 2-4 The kinematics scheme and the CAD model of the robot [Lebosse et al., 2008]

However, the proposed system has some drawbacks. One question that needs to be addressed is the reliability of the redundant mechanism, which provides the manipulator with dexterity and versatility in its motion. There is also no discussion about hybrid position/force control implementation in which this research fails to take force control into account. To date, to the knowledge of the author, there has been no research on force control implementation on TMS robotic system.

2.5.2 A robotized TMS for motion navigated brain stimulation and brain mapping

Matthäus et al. [2006b] and Finke et al. [2008] propose a robotic TMS system for motion compensated navigated brain stimulation and brain mapping system. Their system comprises six main components which are the circular coil, Adept Viper s850 robot, PC (2.8GHz), Polaris tracking system, robot controller and headband as shown in Figure 2-5**.**

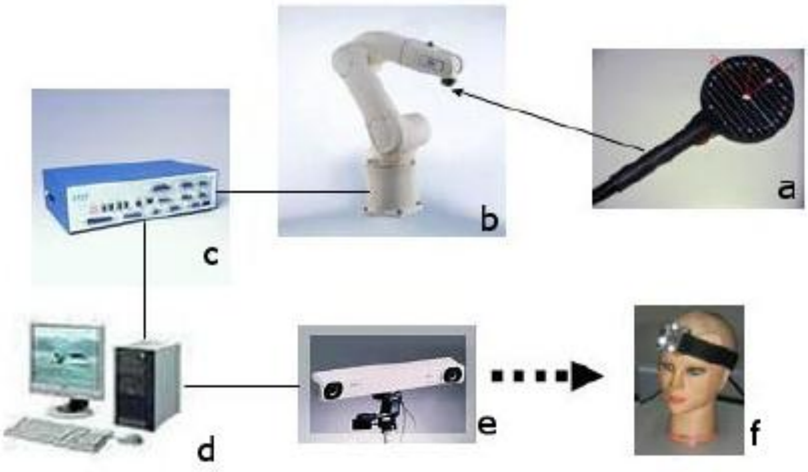


Figure 2-5 The six main components for robot-guided TMS: (a) TMS coil, (b) Adept robot, (c) robot controller, (d) computer, (e) Polaris tracking system, (f) headband [Finke et al., 2008]

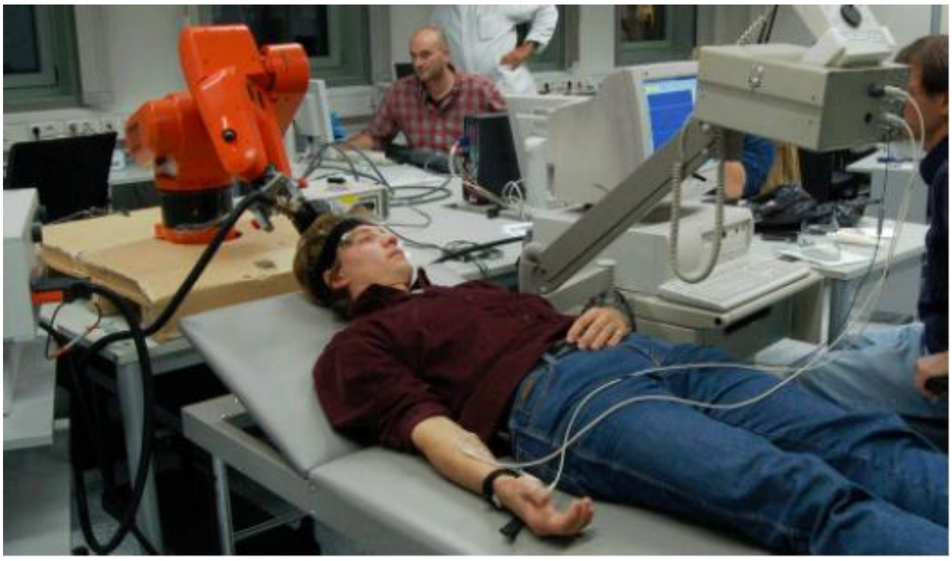


Figure 2-6 Application set up. The patient’s head is tracked by a POLARIS camera (not visible) enabling the robot to compensate for head movements [Matthäus et al., 2006b]

During stimulation, the MRI image data is used to reconstruct the head position and using this information the coil target position is defined. This position is then transformed to robot coordinates using the Polaris tracking system and pre-establish registration. Figure 2-6 shows the application set up of the robot guided TMS system.

During the TMS procedure, a 3D MRI data set is used to create a virtual head model. Then, the virtual world is registered with the real world using the Polaris system. Once the head tracking system establishes a link between the virtual space and the robot workspace, the robot will position the TMS coil according to the path planning and the operator will pick a target point on the subject's head and also the desired distance and orientation of the coil from the head. Nevertheless, approaches of this kind carry with them various well known limitations. The main weakness of the study is the failure to address how the distance between coil and subject's are defined. Consequently, overlooks the fact to quantify the contact force during the procedure. This brings a safety concern if the robot attempts to apply a large force to the subject. Another major source of uncertainty in the study is the system does not take the different coil characteristics into account and the cost of MRI scan as it is necessary to perform the scan for each subject and before the treatment.

Most studies have consistently failed to address the force control as an important feature of the TMS procedure. Most of the research to date has tended to focus on image-guided and navigation system and neglected the study on contact force between the coil and the subject's head which is crucial. It was considered the force control would usefully improve and extend the TMS robotic system. The next section will discuss several force control strategies in order to find the most appropriate force control scheme for the TMS robotic system.

2.6 Review of Force Control Strategies

Robot manipulator arms have been used in industry for many years to perform a variety of tasks involving interaction with their environment in areas such as in manufacturing (assembly, deburring, cutting, polishing and handling parts) and also more recently in non-industrial environment (hospital, welfare, service, maintenance and space) [Werbos, 1974; Pierrot et al., 1999; Poignet et al., 2003]. Implementation of these tasks require a robot to satisfy the specified position trajectory and control the necessary interaction force to either overcome the resistance from the environment or comply with the environment.

In order to achieve successful execution of a number of motions where the robot end effector has to manipulate an object or perform some operation on a surface, control of the interaction between the robot manipulator and its environment is crucial. In a constrained motion, the environment sets constraints on the geometric paths that can be followed by the end-effector. In such a case, the use of a purely position control strategy is not possible unless the tool trajectory is planned extremely accurately and the control ensures a perfect monitoring on this trajectory. The control strategy requires an accurate model of both the robot manipulator (kinematics and dynamics) and the environment (geometry and mechanical features). Manipulator modelling can be known with enough precision, however a detailed description of the environment is difficult to obtain.

In practice, consider an example of a manipulator washing a window with a sponge. The compliance of the sponge may possibly regulate force applied to the window by controlling the position of the end-effector relative to the glass. This approach would work very well if the position of the glass is known very accurately or the sponge is very compliant. However, it is significantly difficult to perform operations in which the manipulator presses against a surface if the stiffness of the end-effector, tool and/or environment is high. If there is uncertainty in the position of the glass or any error in position control, this would ultimately lead to breakage of the glass. This drawback can be overcome by specifying a contact force that is to be maintained normal to the surface. It can be suggested that an enhacement of interaction control performance can be achieved by providing force measurements by mounting the force/torque sensor on a robot manipulator, typically between the wrist and the end-effector and its reading is passed to the robot control unit via a suitable interface [Raibert and Craig, 1981; Burn et al., 2003]. This enhancement is in the form of an outer feedback loop employing compliance or force controller that encloses the inner Cartesian position control system and will be discussed in detail in the following section.

In addition, robot force control involves integration of task goals such as modelling of the environment, position, velocity, and force feedback and adjustment of the applied torque to the robot joints. Zeng et al. [1997] conclude that feedback of various measurement signals of the output of a robot such as position, velocity, force and the selection of command input signals to a robot result in different force control methods.

2.7 Fundamental of Force Control Schemes

When a robot is moving freely along a trajectory, position control alone is able to achieve the desired motion. However, force control will have to be applied if the task involves any contact between the robot end-effector and the environment. The force control application offers enhancement to the effectiveness of position control. Dombre and Khalil [2007] suggests that robot force control algorithms can be divided into several methods:

- involving the relation between position and applied force – active stiffness control;
- using the relation between velocity and applied force – impedance control;
- using position and force feedback - parallel hybrid position/force control and external force control;
- using force feedback – explicit force control;

The following sections will discuss these control methods in details.

2.7.1 Active stiffness control

Salisbury [1980] proposed a method to control the apparent stiffness of a manipulator called *active stiffness control*. In this approach the robot is considered as a programmable spring, where the compliant behaviour of the manipulator can be controlled by changing stiffnesses under program control to achieve varying task requirements. This allows the programmer to specify the three translational and three rotational stiffness of a frame located arbitrarily in task space coordinates.

Figure 2-7 illustrates the block diagram of active stiffness control. For a better understanding this control scheme can be divided into two parts, namely the controller and the basic dynamic system.

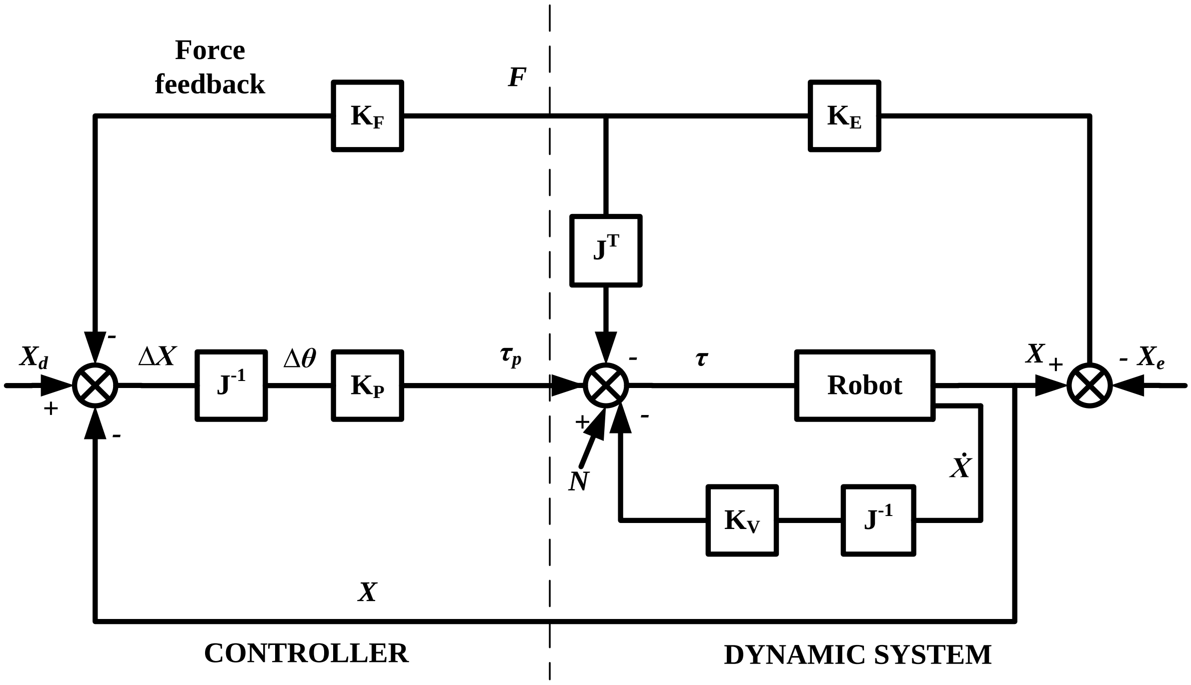


Figure 2-7 Active Stiffness Control scheme

where the variables can be defined as listed;

J	Jacobian matrix
X_d	the desired position in task space
X	Position in the task space
$\dot{X}$	Velocity in the task space
ΔX	Position error
$\Delta\theta$	Joint angle displacement
τ_p	Joint torque
N	Feedforward compensation for gravity, centrifugal and Corialis forces
X_e	Position of the contact environment
$\mathrm{K_E}$	Stiffness of the sensor and environment
$\mathrm{K_p}$	Joint stiffness matrix
$\mathrm{K_F}$	Compliance matrix
$\mathrm{K_v}$	Velocity damping matrix
F	Contact force

The basic dynamic system comprises a robot and its environment as well as velocity feedback and nonlinear compensation for linearizing the robot dynamic system. The basic stiffness formulation ($F=K\Delta X$) is applied for the stiffness control loop implementation, where K is the Cartesian stiffness matrix. Using the Jacobian matrix differential transform, the Cartesian displacement ΔX is determined from the joint angle displacement $\Delta\theta$ in the form of

$$\Delta X = \mathrm{J}\,\Delta\theta \tag{2-2}$$

Assuming that the friction and dynamic forces are compensated or are small enough to be neglected, the joint torque τ that is necessary to apply the contact force is computed using

$$\tau = \mathrm{J}^{\mathrm{T}}\,F = \mathrm{J}^{\mathrm{T}}\,\mathrm{K}\,\Delta X \tag{2-3}$$

Combining equations (2-2) and (2-3),

$$\tau = \mathrm{J}^{\mathrm{T}}\,\mathrm{K}\,\mathrm{J}\,\Delta\theta = \mathrm{K_p}\,\Delta\theta \tag{2-4}$$

Thus, the commanded joint torque τ_p due to the contact force is defined as follows

$$\tau_p = \mathrm{K_p}\,\Delta\theta \tag{2-5}$$

where K_p is the (6x6) joint stiffness matrix. This approach allows the user to specify mechanical stiffness of the manipulator by varying the value of K_p which determines the proportional gains of the stiffness control loop in the joint space. This makes the robot stiff along the axis under position control so that it can follow the desired position trajectory as close as possible, while a low gain is set to force controlled directions to prevent excessive build up of contact force. One of the limitation of this control scheme is the control can be lost at singularities since the Cartesian stiffness matrix is transformed using Jacobian [Kiguchi and Fukuda, 1999].

2.7.2 Impedance control

According to Hogan [Hogan, 1985a; Hogan, 1985b], the fundamental philosophy of impedance control is that the manipulator control system should be designed not to track a motion trajectory alone but rather to regulate the mechanical impedance Z_m of the manipulator, as defined by the relationship between the velocity $\dot{X}$ and the applied force *F*. In the frequency domain, this is represented by

$$\frac{F(s)}{\dot{X}(s)} = Z_m(s) \tag{2-6}$$

In terms of position *X(s)*, the equation is

$$\frac{F(s)}{X(s)} = sZ_m(s) \tag{2-7}$$

In general, the robot is equivalent to a second order mass-spring-damper system, with transfer function;

$$sZ(s) = Ms^2 + Cs + K \tag{2-8}$$

where *M*, *C* and *K* represent the desired inertia, damping and stiffness matrices respectively. There are two ways to implement the impedance control behaviour which are depending on the need of an external force measurement [Dombre and Khalil, 2007] as shown in the Figure 2-8 and Figure 2-9;

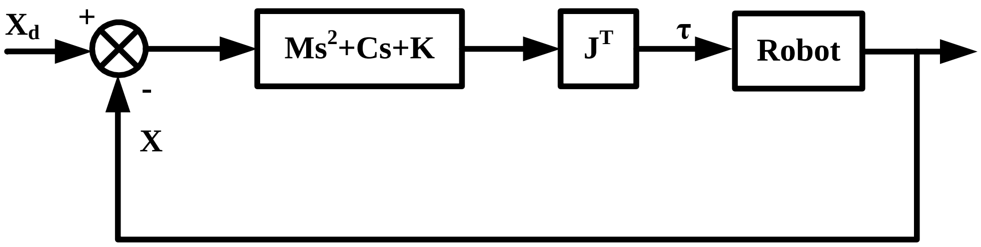


Figure 2-8 Impedance control scheme without force feedback

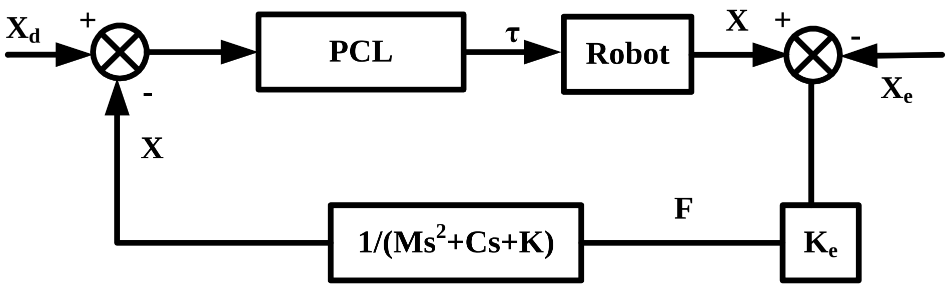


Figure 2-9 Impedance control scheme with force feedback (PCL: position control loop)

Figure 2-8 illustrates an implementation of the impedance control strategy without force feedback or so called implicit force control. The dynamics of the robot is neglected except for the gravity compensation Q. In this case, the control law is given by;

$$\tau = J^T[B(\dot{X}_d - \dot{X}) + K(X_d - X)]\dot{\mathrm{X}} + Q \qquad \text{(2-9)}$$

The K and C matrices are equivalent with the proportional and derivative gains in the task space (PD control) which can be represented as the stiffness matrix and the damping matrix of the robot respectively. This type of control is suitable during robot small motions for example in mechanical parts insertion task.

The second method can be derived from the dynamic robot model in the joint space which yields the following equation

$$\tau = M(q)\ddot{q} + H(q, \dot{q}) + J^T F \qquad \text{(2-10)}$$

where M is the inertia matrix of the robot, and H contains the Coriolis, centrifugal and gravity forces/torques. By expressing the vector $\ddot{X} = J\ddot{q} + \dot{J}\dot{q}$ with joint variables in equation (2-10) yields to

$$\tau = M(q)J^{-1}[\ddot{X} + \dot{J}\dot{q}] + H(q, \dot{q}) + J^T F \qquad \text{(2-11)}$$

then replacing the vector $\ddot{X}$ as expressing in the following equation;

$$\ddot{X} = \ddot{X}_d + M^{-1}[B(\dot{X}_d - \dot{X}) + K(X_d - X) - F] \qquad \text{(2-12)}$$

which leads to

$$\tau = \ddot{J}^{-1}[X_d + M^{-1}\left[B\left(\dot{X}_d - \dot{X}\right) + K(X_d - X) - \dot{J}\dot{q}\right] + \ H(q, \dot{q}) + J^T F \qquad (2\text{-}13)$$

The above derived formulation requires an external force measurement as well as a complete calculation of the estimated dynamic model. This type of control scheme has similarity with active stiffness control in two aspects, namely the sensed force is calculated by the position modification, and control of joint torques.

2.7.3 Explicit force control

A theoretical and experimental investigation of an explicit force control strategy was reported by Volpe and Khosla [1993]. The explicit force control involves the direct command and measurement of force values with the aim of having the output as close as possible with the desired input. Figure 2-10 illustrates schematic diagram of force based robot explicit force control. In this type of control scheme the measured force is directly used for feedback in order to form the force error vector. The force control law is normally chosen as one of the subsets of PID control.

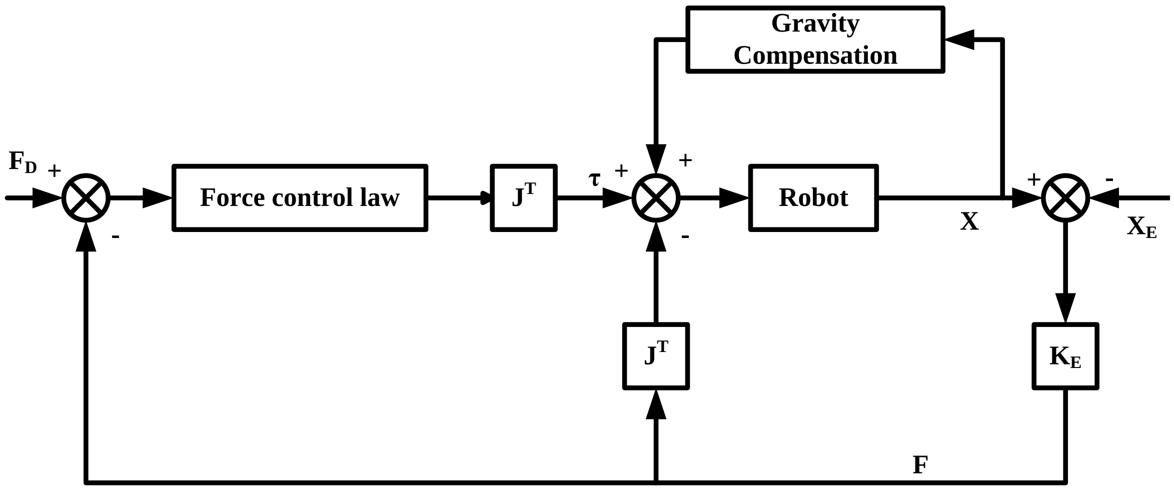


Figure 2-10 Explicit Force Control

Volpe and Khosla's [1993] study suggests that the best force trajectory tracking is achieved with integral gain. The study demonstrates that PD (proportional-derivative) force control or damping strategies are not promising method to provide stability to the system during contact with the environment. The framework of their studies provides the greater understanding of stability is affected by gain variations. See the reference [Volpe and Khosla, 1993] for further discussion.

2.7.4 Hybrid position/force control

Railbert et al.[1981] proposed the most well-known approach to compliant motion called hybrid position/force control. Numerous researches have been reported on this subject [Raibert and Craig, 1981; Khatib, 1987; Whitney, 1987; Degoulange and Dauchez, 1994]. The idea behind this control scheme is an architectural concept that associates the constraints of a task requiring force feedback to the controller design. In hybrid position/force control the position and force are separately controlled as shown in Figure 2-11 below.

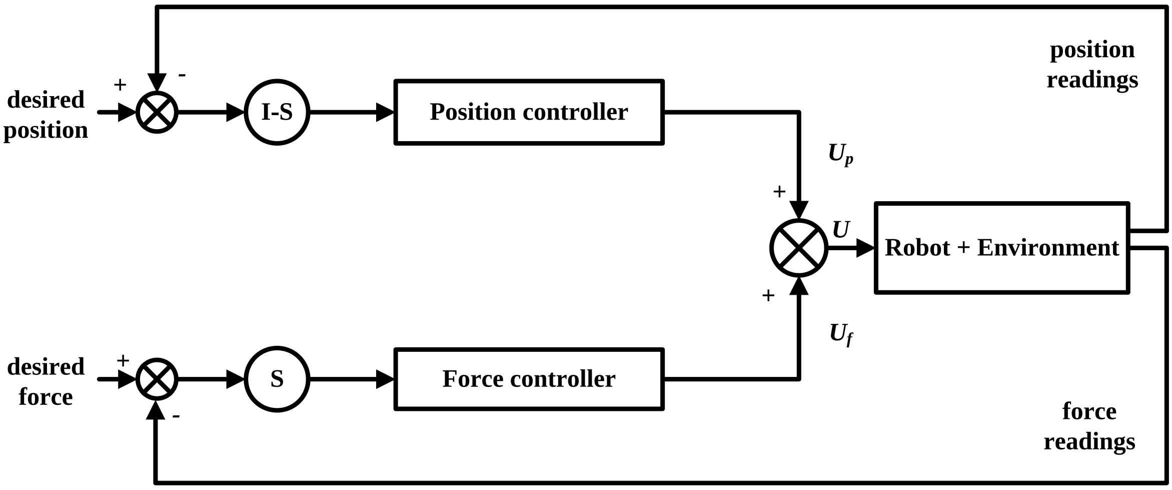


Figure 2-11 Hybrid position/force control

In Figure 2-11 S = diag (s_j) ($j = 1 \ldots n$) is called the compliance selection matrix and the number of degree of freedom is defined by n. The subspaces for which force or position are to be controlled is determined using this matrix S. When $s_j = 1$, the jth DOF must be force-controlled, otherwise it must be position-controlled. As shown in Figure 2-11 the hybrid controller consists of two complementary sets of position and force feedback loops controlling the manipulator arm. Normally, the position servo loop is already implemented in the original robot controller and the force control loop must be implemented with an action at the joint torque level. It is important to ensure orthogonality between the outputs of the force and position controllers to prevent any conflict at the actuator level. Furthermore, if disturbances occur along the force-controlled direction, the induced positioning cannot be corrected.

A comparative study by Perdereau and Drouin's [1993] showed that determining the position and force controller gains is very difficult as a result of the burden of calculations to determine equations of dynamic force and position responses which cannot be simplified. A number of variants of the original hybrid scheme have been developed, most to reduce computational burden [Perdereau and Drouin, 1993; Kiguchi and Fukuda, 1997; Kiguchi and Fukuda, 1999; Jantzen, 2007]. Moreover, this characteristic equation is totally non-linear because it depends not only on the mechanical structure but also on the task configuration by the selection matrix *S* as well as the operating point through the Jacobian matrix *J*. It is important to keep the servo rate as high as possible to accomplish good position and force control. Ow [1998] concludes that experimental implementation of computed torque system such as hybrid position/force control has disappointing low performance, often with hesitant and jerking motion. This undesireable performance is caused by poor rejection of disturbance torque which arise from a wide variety of sources. Refer to [Lin and Huang, 1998] for further discussion on disturbance rejection in order to resolve the problem.

2.7.5 External force control

The '*external force control scheme'* developed by De Schutter et al. [De Schutter and Van Brussel, 1988a; De Schutter and Van Brussel, 1988b] is considered to offer a better solution regarding the safety constraints, simplicity and implementation efficiency [De Schutter and Van Brussel, 1988a; De Schutter and Van Brussel, 1988b; Degoulange et al., 1993a; Degoulange et al., 1993b; Perdereau and Drouin, 1993; Degoulange and Dauchez, 1994; Ow, 1997; Pierrot et al., 1999; Pierrot et al., 2000]. The key advantages of this control scheme can be summarized as follows:

- Stability-switching between the position loop and force loop is avoided.
- Uses a simple and reliable control law (such as PID) for robust control.
- Easy to implement on most types of controller and robot.
- Handle both position and force control information in the same Cartesian direction.

- Suitable for safety requirement applications because this type of control scheme does not present the drawback of kinematic instability and guarantees that all directions in space are always fully controlled.

This method is composed of two embedded control loops; an outer force control loop and an inner position control loop as illustrated in Figure 2-12. The output of the outer loop ΔX_F is transformed into a desired position input X_R for the inner loop.

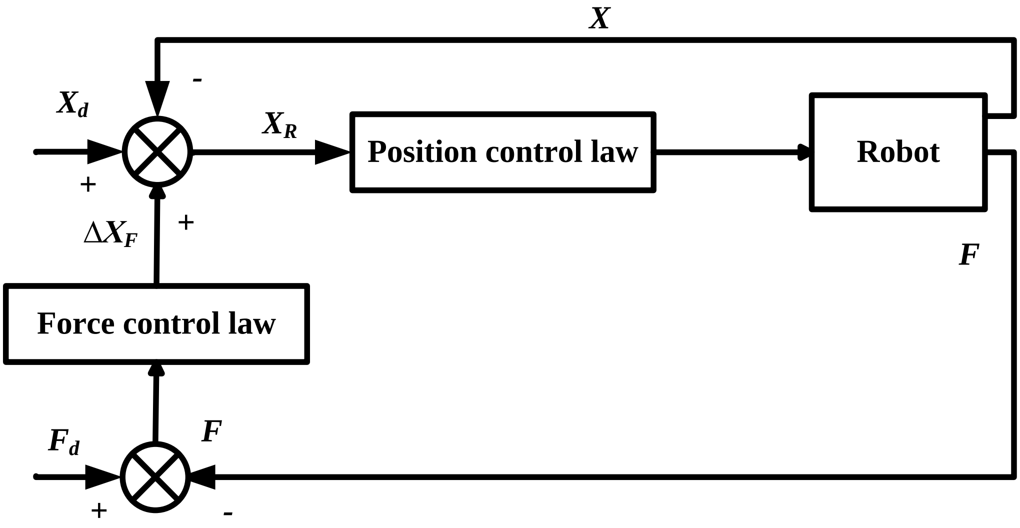


Figure 2-12 the external force control

The main advantage of controlling force using an external control loop, whilst still maintaining an inner position loop, is in the relative ease of implementation since it does not require much modification to the existing robot control system, since the force control law is defined in Cartesian coordinate and it is possible to input the position control law (original robot controller) with such a Cartesian position/orientation. In addition, simultaneous control of position and force is easy to achieve with force being controlled by modifying the robot position demand.

2.8 Recent Application of External Force Control on Medical Robotic System

Prior to implementation of an external force control in TMS robotic system, an application of this control strategy on several medical robotic systems are discussed in this section.

2.8.1 Hippocrate

Pierrot et al. [1999] developed a robotic system to assist doctors to move ultrasonic probes on the patient's skin while exerting a given effort. Figure 2-13 illustrates the *Hippocrate* robotic system consists of Mitsubishi PA-10 robot arm with low level controller, a PC running a real-time OS for the high level controller, a dead man switch, a teach pendant and a force/torque sensor mounted at the end-effector. From their feasibility study they evaluated several force control schemes on a large variety of robots including single or dual arm serial robots, parallel robot and redundant robot and concluded that the external force control scheme for example in Perdereau and Droin [1993] is the best solution regarding to simplicity, safety and implementation efficiency.

Figure 2-14 provides a schematic diagram for external force control scheme that implemented in *Hippocrate* robotic system. As can been seen from Figure 2-14, the robot joints are always controlled in position ('position control in joint space' block), depending only on measurements of actual robot joint positions q_r. This low level position control block receives inputs from the 'inverse kinematics model' block, which transforms the Cartesian position (i.e. the probe position, P) into joint positions. The external force control loop computes the force error resulting from the force acting on the robot tip and the gravity compensation module where a wrist-mounted force sensor is used to measure the actual force and a software module is written to compensate the probe weight. A selection matrix validates the modified position by the force control loop. The output of the force control loop is used to alter the desired Cartesian provided by the trajectory generation module.

In contrast to Perdereau and Droin [1993] external force control scheme, the selection matrices is introduced in the external control loop to provide a unique scheme for both position and force control. Practically the position information is used to monitor the robot while the light contact force (3N) is controlled only in the normal direction to the skin. These research findings provide a new understanding of industrial robot capabilities with an appropriate force control to improve medical techniques particularly in applications that involved precise and accurate measurement as well as repeatability procedure.

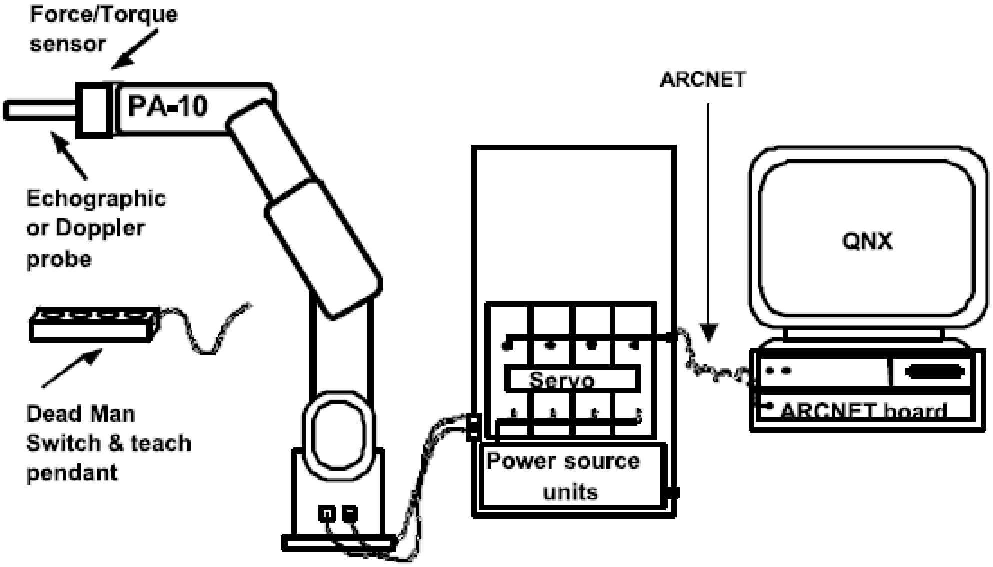


Figure 2-13 *Hippocrate* system used for feasibility study [Pierrot et al., 1999]

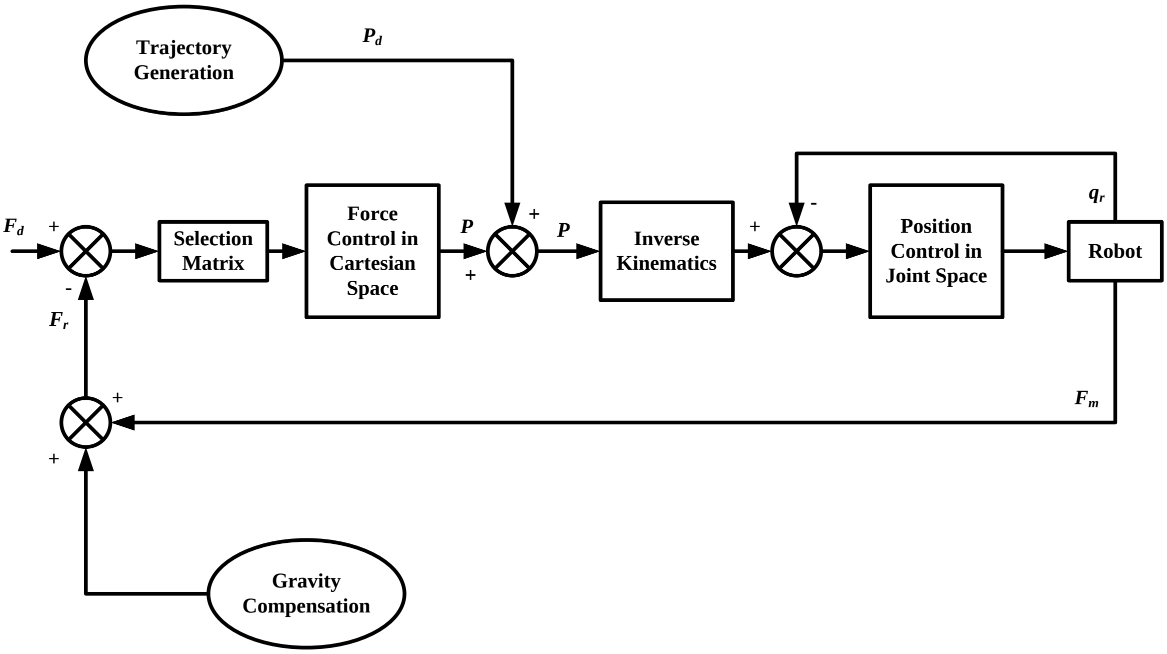


Figure 2-14 Schematic diagram for external force control scheme

2.8.2 Dermarob

A novel and safe robotic system for reconstructive surgery (skin harvesting) was constructed by Dombre et al. [2003]. In contrast to *Hippocrate* which used an existing commercial industrial robot, an innovative and re-engineered robot with SCARA (Selective Compliant Articulated Robot Arm) architecture with nonsphererical wrist was designed due to safety requirement for skin harvesting procedure as shown in Figure 2-15. A feasibility study [Pierrot et al., 2000] was undertaken to investigate surgeon motion in order to design and implement an ideal position/force control scheme.

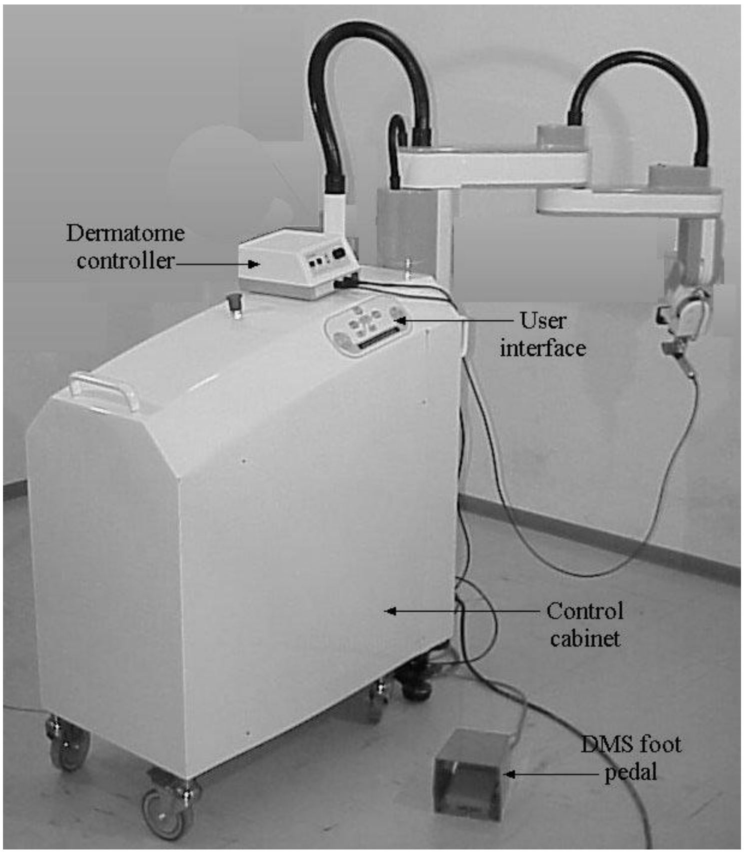


Figure 2-15 Dermarob and control cabinet [Dombre et al., 2003]

In *Dermarob*, external control with Cartesian summation between force and position components as well as joint control by Inverse Geometric Model (IGM) has been adopted. The block diagram of Figure 2-16 illustrates how the controller works [Dombre et al., 2003]. After appropriate transformation, the difference between the desired and measured forces generates an increment which is used to modify the desired motion computed by the motion generator. S is the selection matrix that allows the surgeon to select the Cartesian directions along or about which the force and moment will be controlled. The closed-form solution of the IGM computes the joint positions sent to the axis control board (PMAC-PC board from Delta Tau).

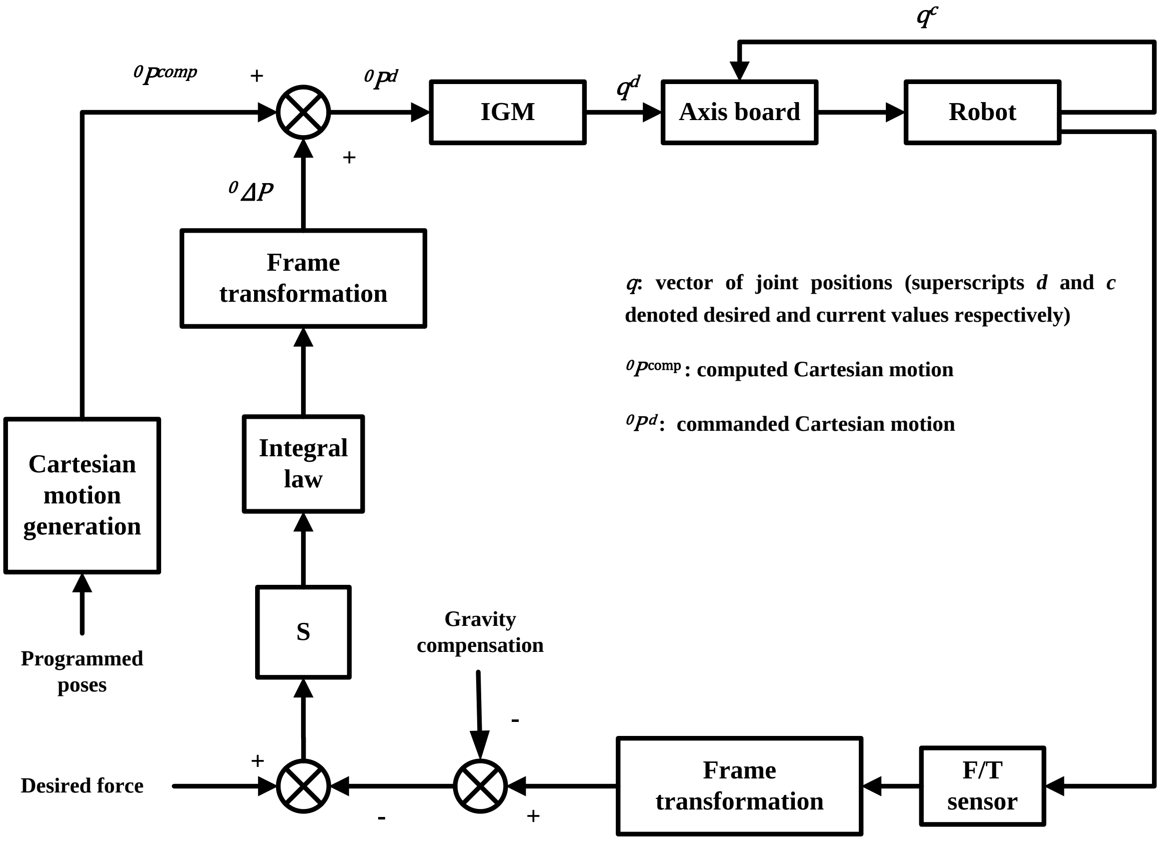

q^c
$^0P^{comp}$
+
$^0P^d$
IGM
q^d
Axis board
Robot
+
$^0\Delta P$
Frame transformation
q: vector of joint positions (superscripts d and c denoted desired and current values respectively)
$^0P^{comp}$: computed Cartesian motion
$^0P^d$: commanded Cartesian motion
Cartesian motion generation
Integral law
S
Gravity compensation
Programmed poses
-
+
Desired force
-
+
Frame transformation
F/T sensor

Figure 2-16 The external force control scheme implemented in *Dermarob* robotic system

To conclude, an active force controller is mandatory in order to apply a desired force on the patient. *Dermarob*'s control strategy is similar to that used in *Hippocrate*, although it is different in a number of respects. *Hippocrate* maintained a constant light force on the defining position on the human skin. On the other hand the desired force applied by the *Dermarob* is much higher which deform the skin along a continuous path.

2.9 Summary

Although the TMS technique is very significant, it is not yet widely accepted because of the observed inconsistency in efficiency between subjects. TMS coils produce very dense and defined field, which mean that a small movement of the coil position leads to a substantial change of the electric field delivered to a target nerve. It is due to the difficulty to indicate correct positioning with the presently available stimulation systems which leads to poor repeatability. This explains the vital need for technologies to support precise navigation and positioning of TMS coils.

Several researchers have developed image-guided TMS to overcome this problem [Herwig et al., 2001; Lancaster et al., 2004; Matthäus et al., 2006b; Finke et al., 2008; Lebosse et al., 2008]. By registering MRI data with the patient’s head and by tracking the coil, the area of stimulation can easily be determined (Ettinger et al., 1998) [Ettinger et al., 1998]. Interleaving TMS and functional brain imaging offers much promise. Matthäus et al. [2006b] pointed out a major problem with this kind of technique; exact stimulation is hard to achieve at a pre-defined point, particularly if the site is to be stimulated several times or continuously as in the treatment of depression. Either head movements must be followed by manually adjusting the coil continuously or the head must be fixed. George et al. [1999] discuss two problems for combining TMS with functional and structural neuroimaging. One of the limitations with this technique is that sometimes it is difficult to match the imaging technique to the temporal duration of TMS. Additional problems have concerned the interference produced by TMS with image acquisition.

So far, all the research reviewed to date is limited in one way or another. Most studies used the robot arm to position the coil manually. It is becoming increasingly difficult to ignore the developments and methods of safe and reliable algorithms in TMS robotic system. The system should be designed as a concurrent process or be real-time controlled to ensure complete confidence in the subject’s safety as well as the reliability of the system. Another problem with the above approaches is that MRI and fMRI images are used to build a 3D model of the brain so the stimulated cortical regions can be specified. However, the study fails to consider that an image-guided method is costly and is accompanied with the burden of computational calculation. In addition, exact stimulation at a predefined point is hard to achieve, particularly if the site is to be stimulated several times or continuously as in some treatments.

Despite its safety and efficacy, current researches have consistently failed to address the force control as an important feature of the TMS procedure. Most of the research to date has tended to focus on image-guided and navigation system and neglected the study on contact force between the coil and the subject's head which is crucial. In the TMS robotic system, the force control has not yet been demonstrated, thus this leads to this work which is presently addressing the issue.

Robot force control can be classified into several categories depending on the selection of output signals and command input signals. All the studies review so far provide an idea and concept of force control implementation in various tasks. However, with a different application, different strategies must be applied depending on the requirement in order to achieve satisfy performance. Bear in mind that the built-in commercial robot position controller also plays a vital role in defining suitable scheme, as direct access to actuator level is restricted. The external force control composed of an inner position control loop and outer force feedback loop is found as an appropriate control scheme to apply in TMS robotic system. Furthermore, several successful implementation external force control in medical application were reviewed to provide an idea and concept for further development in force- controlled TMS robotic system.

CHAPTER 3

TMS PROCEDURE TASK ANALYSIS

The goal of the study is to automate the TMS procedure in order to satisfy two requirements, namely to maintain the hot spot position to achieve a better localization during treatment, and to maintain a constant contact force between the coil and the human's head. Prior to the implementation of force control to the TMS robotic system, an analysis was carried out to identify the procedure requirement related to the application of force control. It was found that it is extremely hard to mimic a neurologist's gesture during the TMS procedure particularly in terms of kinematics and force. The motion can be described in 3D space and thus involves quite a large force exerted on the subject's head. Basically, the subject's head surface is very difficult to model as it depends on the skull bones, hair volume and tissues underneath. Furthermore, the subject's head movement is also unconstrained during the TMS procedure which can introduce problematic issue for force control application. Thus, the kinematic analysis of the head-neck system was conducted and will be described in detail in section 3.1.1. On the other hand, the force and motion produced by the neurologist rely on various parameters such as patient's head, fatigue as well as neurologist's skill. There is no evidence that two neurologists will conduct the procedure with neither the same motion nor the same force.

3.1 TMS task specifications

The TMS procedure can be divided into three successive steps as illustrated in Figure 3-1. The neurologist begins to move the coil from the start position to the target position along *x* axis (step 1). Once the target position has been reached, the procedure begins by pressing the coil to the human head along *z* axis which normal to the target head position (step 2) and finally withdraws the coil from the contact (step 3) once the treatment is completed.

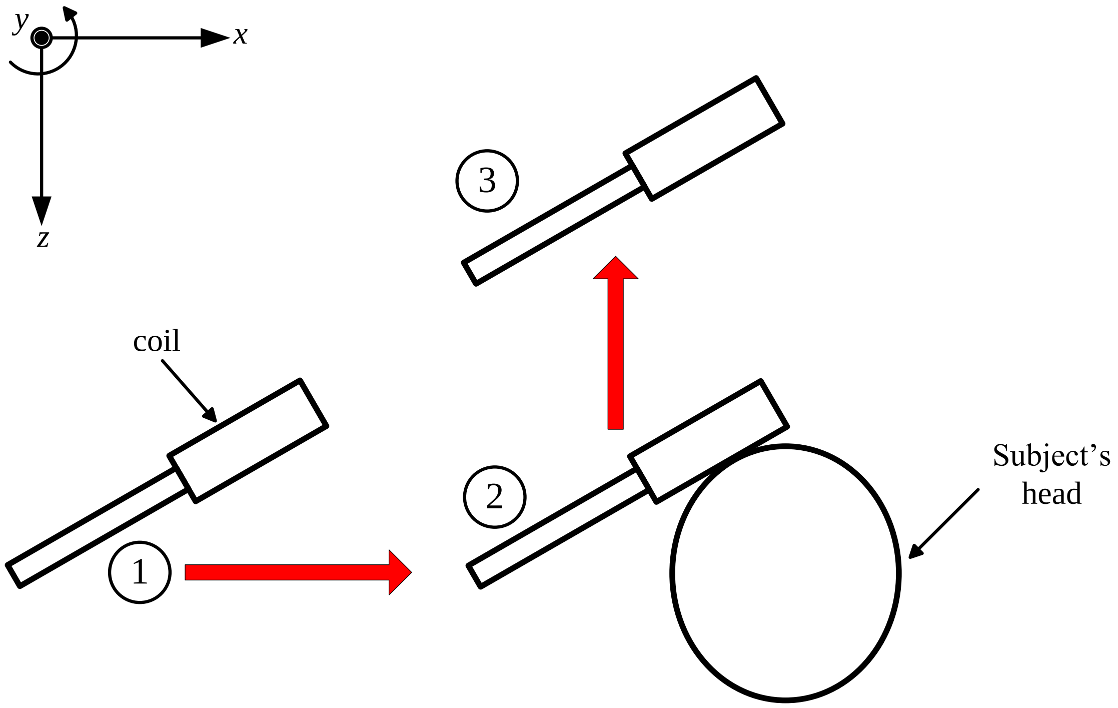


Figure 3-1 Two main steps of the TMS procedure

In the context of force control strategy, the first step can be considered as free space movement or a guarded move (preoperative phase), while the second step (intraoperative phase) involves a process to maintain a force which required to be maintained normal to the subject's head surface (constrained space movement). The details of the mentioned movements are described in the following section.

3.1.1 Free/unconstrained space movement (step 1 and 3) – Preoperative phase

The assumptions of free space movement have been addressed in previous studies and can be concluded that the free space movement is purely position control where the robot moves at low speed until predetermined force level is detected. De Schutter and Van Brussel [1988b] referred this force control strategy as a 'guarded move'. Besides, the free space movement can also be achieved by tele-operation or haptic guidance in which the entire axes are force-controlled [Dombre et al., 2003; Po-ngaen, 2006]. Using this method, the robot arm moves at velocity proportional to the applied force.

3.1.2 Constrained space movement (step 2) – Intraoperative mode

In order to achieve a robust force control scheme for compliant motion task, the formalisation approach proposed by Mason [1981] is used in this analysis. This will serve as the interface between the manipulator with the appropriate control strategy. The approach is based on situations in which motion of the manipulator is partially constrained due to contact with the surface environment. Given that all other parameters remain constant including the coil and robot end-effector attachment, the robot speed, environment properties and compliance, then the TMS procedure (step 2) can be controlled by maintaining a constant normal contact force F_N and position tangential x_T to constraint surface, as shown in Figure 3-2.

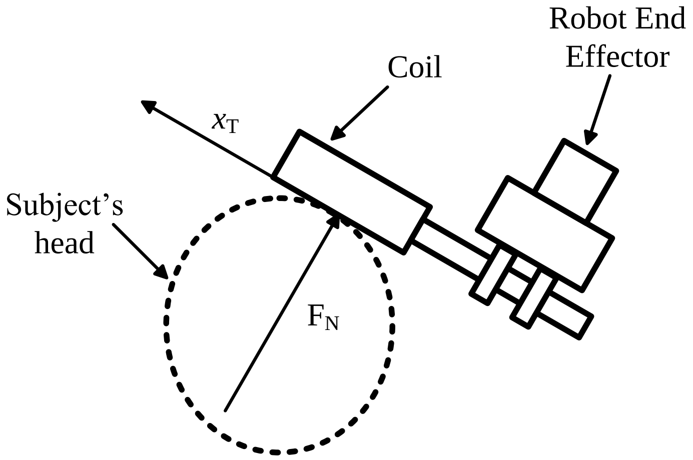


Figure 3-2 The contact forces applied to the subject's head

This contacting surface is defined as a constraint surface (C-surface) in task space. With reference to Figure 3-2, motion is only possible along the tangents to the C-surface and the natural force is unconstrained along the C-surface normal. Consequently, the manipulator is not free to move through the surface as a natural position constraint exists in this direction. These two natural constraints (position and force) illustrate the task limitation and divide the manipulator degrees of freedom into two orthogonal sets, one dealing with the position and other with the force. The formalization presents the task description which contains all the information required in order to allow an entirely automatic execution of the task.

3.2 Defining the Problem

3.2.1 Contact force during conventional hand-held TMS procedure

Having established the fundamental control requirements of TMS task, the previous analysis indicates that a force control is desirable in TMS robotic system, especially when unforeseen contact force arise. A simple test was carried out to identify the applied contact force during a hand-held TMS procedure. The measurement involves two positions which are approximately perpendicular to human coronal plane and 30~45 degrees to the head surface respectively, as illustrated in Figure 3-3.

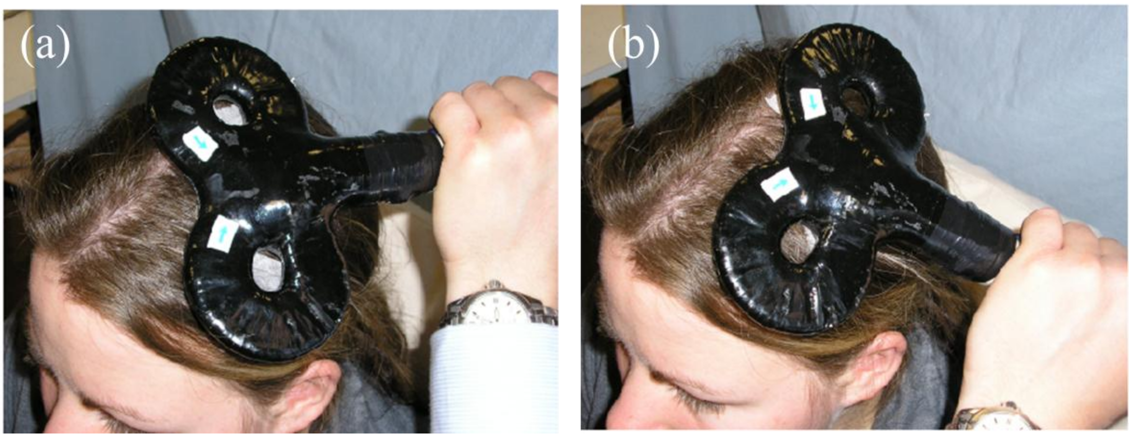


Figure 3-3 Two different target positions a) perpendicular to human coronal plane and b) approximately 45 degrees to the head surface

Ten voluntary participants were recruited and selected randomly for this test. The subjects were asked to sit on the chair as comfortable as possible and were allowed to make small movements during the test. Prior to commencing the test, a force/torque sensor is attached to the TMS coil as shown in Figure 3-4. The applied contact force was recorded using F/T controller for further analysis. The test was repeated 3 times for each participants and each test was completed in approximately 20 minutes.

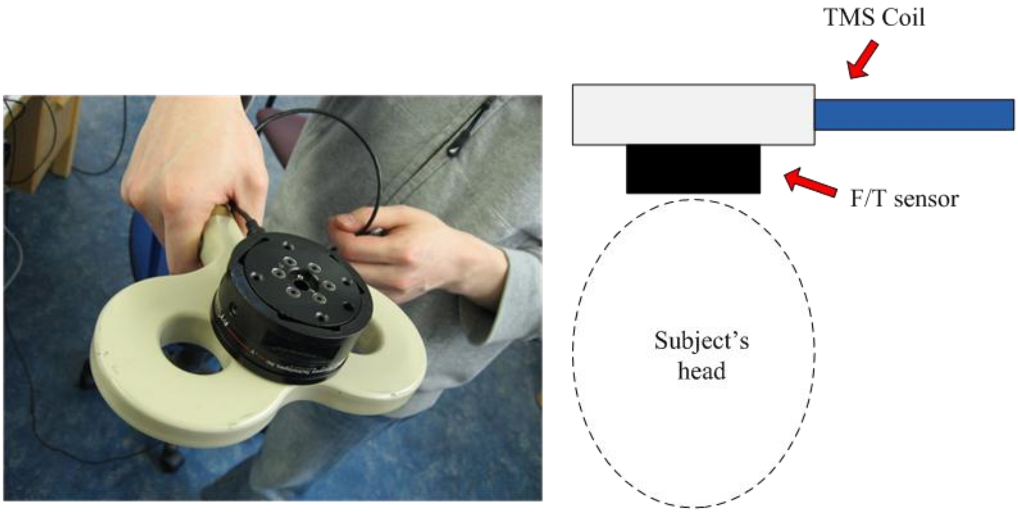


Figure 3-4 TMS coil and force/torque sensor configuration

In addition, the test was also repeated on a ball which used to represent human's head for this study. Figure 3-5 presents the picture of the ball which is attached to a 6-axis PUMA-560 robot arm to simulate human's head movement. As safety is the main concern of this study, the ball-PUMA 560 robot is adopted for further experimental tests in following chapters.

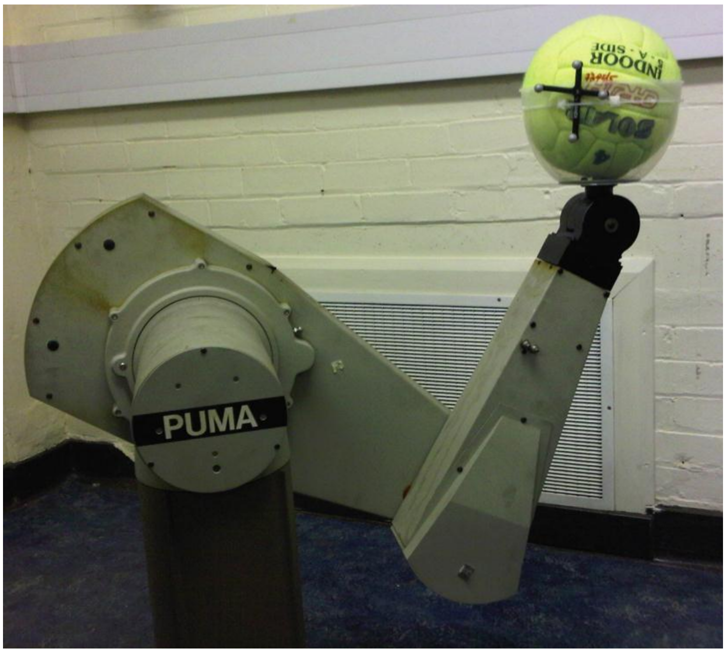


Figure 3-5 Ball-PUMA 560 robot configuration

3.2.1.1 *Preliminary test results*

Figure 3-6 and Figure 3-7 provides force/torque measurements when the coil is positioned perpendicular to human coronal plane (vertical orientation). As can be seen from the graph, a contact force (approximately 6N) is sensed along *z* axis, the direction normal to subject's head surface. The force and torque values on other direction are nearly zero as expected.

An observation made during the test was the contact force/torque measurement is considerably varied and inconsistent with different subjects and even with a repetitive and similar procedure (Figure 3-1), showing that it is difficult to maintain a constant force during the same procedure. This could be attributed to the inability of the neurologist to hold and maintain the coil on the subject's head. Furthermore, this becomes more problematic as subject's head characteristics (i.e. hair volume, skull, tissue) are varied from one to another.

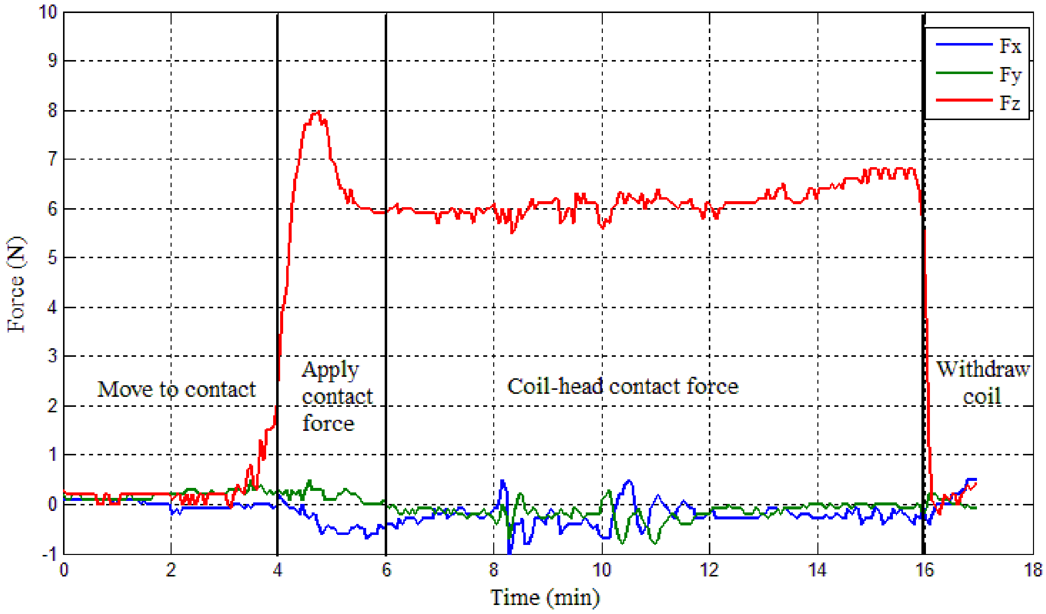


Figure 3-6 Applied contact force along *x*, *y* and *z* axes (vertical orientation)

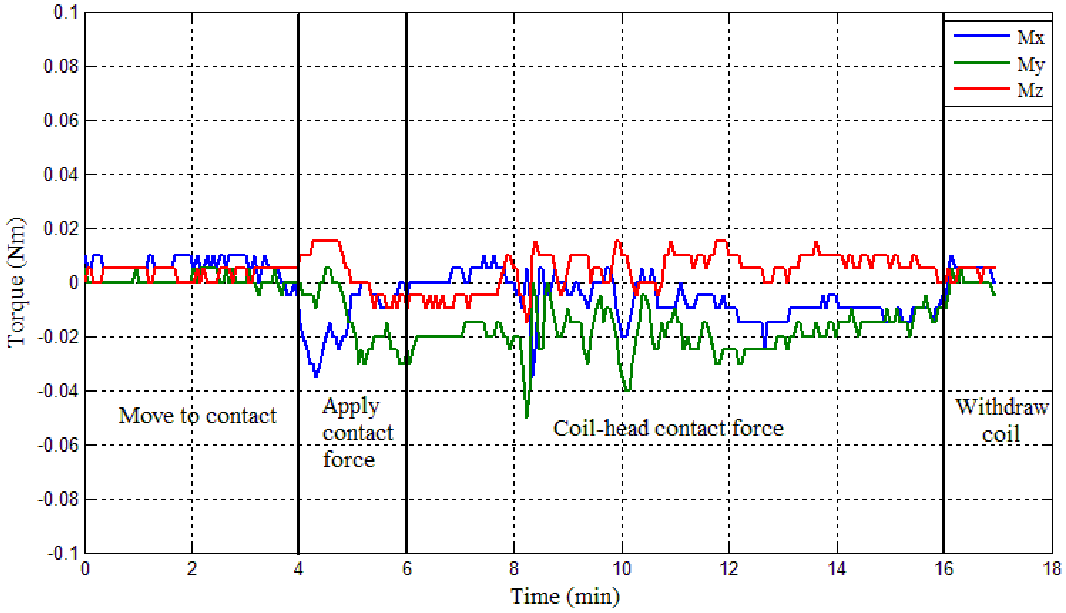


Figure 3-7 Applied contact torque along *x*, *y* and *z* axes (vertical orientation)

For the measured contact force at second defined position (30~45 degrees to the head surface), the same explanation applies and the test results (Figure 3-8 and Figure 3-9) are similar to those shown in Figure 3-6 and Figure 3-7. It also noticeable that a large contact force is sensed on first contact with the subject, resulting in slightly uncomfortable situation.

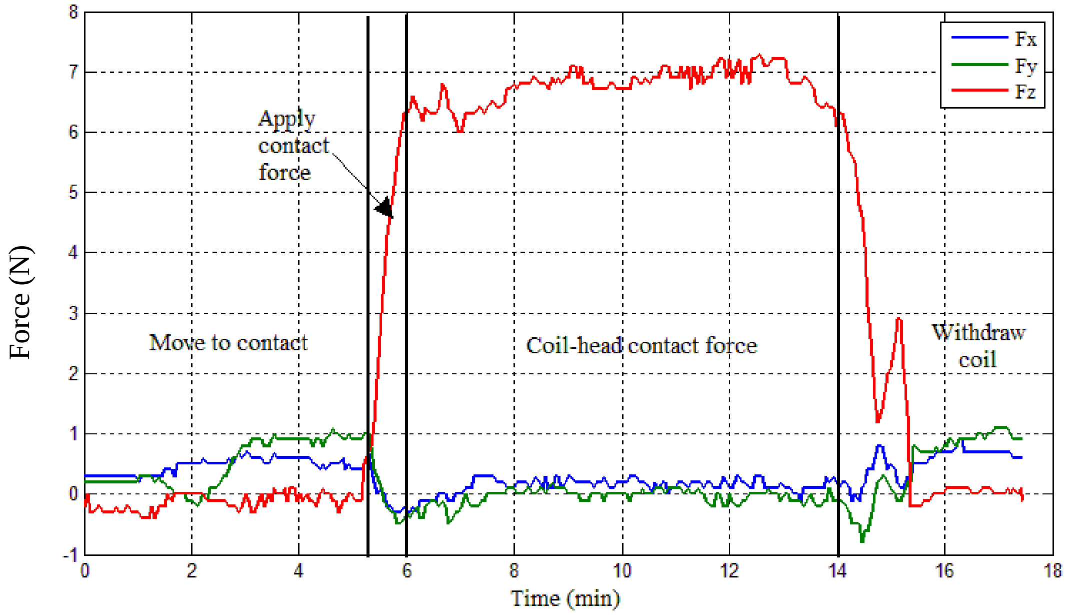


Figure 3-8 Applied contact force along *x*, *y* and *z* axes (~30 degrees orientation)

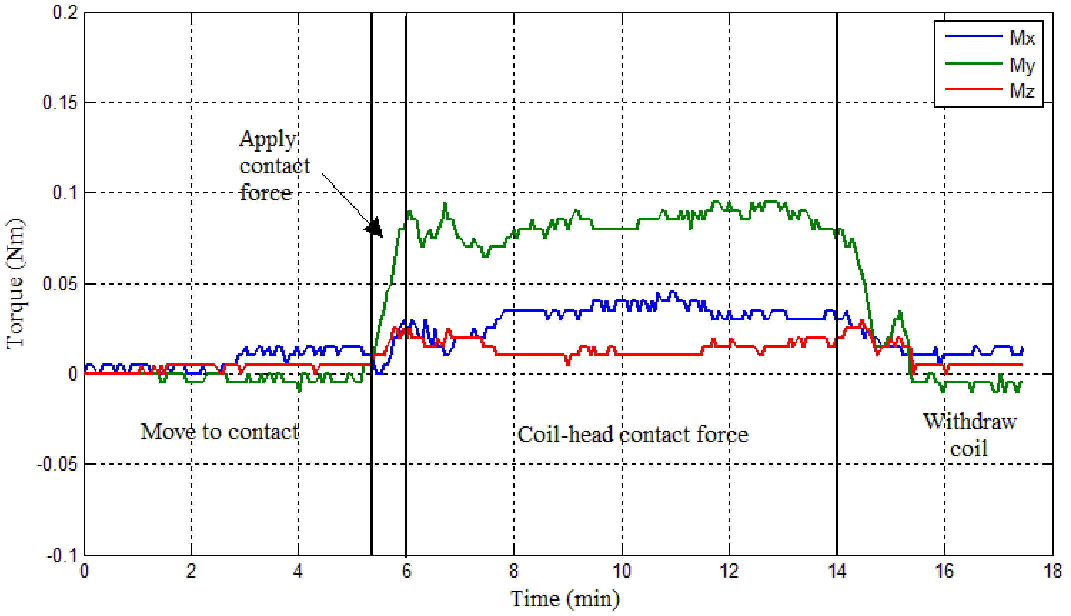


Figure 3-9 Applied contact torque along *x*, *y* and *z* axes (~30degrees orientation)

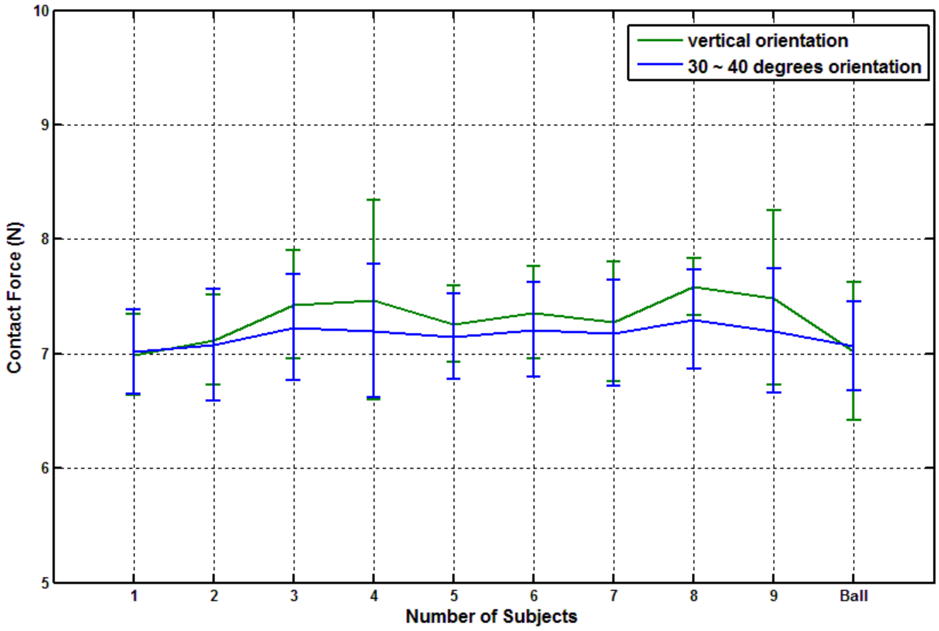


Figure 3-10 TMS hand-held procedure test results

Figure 3-10 presents the comparison of measured force results for both the human's subjects (head) and the ball. The values averaged during repeated tests for all participants. In all cases the average applied contact force is approximately 7N and is kept roughly constant against the subject’s head during the procedure. The applied contact force on the ball is found similar to that measured on the human’s subject head.

The test results are subjective and imply that it is almost impossible to apply a constant force during the hand-held TMS procedure. Inconsistent contact force measurement was found when the procedure is repeated even though the operator, coil and subject are same. This scenario can be described as problems that can subsequently jeopardize the TMS treatment effect. It was decided that the best solution can be achieved by automating the procedure using robotic technology and an appropriate force control scheme to improve the procedure.

Taken together, the evidence from the test also suggests that the contact force can be controlled in one direction which is normal to the subject’s head surface during constrained task while the other directions are position control. For preoperative phase, the task can be carried out in two ways either it is fully automatic control in which the robot is program to move to the desired position or an alternative strategy that is to resort to tele-operation mode which remains one of the most reliable ways of carrying out the task on the unknown properties environment (subject’s head). Following section will discuss the unknown human head properties particularly unconstrained head motion during the treatment.

3.3 TMS Task Challenge

During conventional hand-held procedure test, the following issues were observed and can be considered as challenges to the force-controlled TMS robot system implementation;

1. The hot spot position is not directly measured and would need to measure and identify first prior to the treatment.
2. The applied force exerted by the operator along normal direction to the subject’s head surface was found ranging from 6N to 10N in which the operator keeps the coil in close contact to the hot-spot position.
3. The subject’s head geometrical, inertial and structural stiffness properties are totally unknown and vary from one subject to other subjects.

4. During the treatment, even the subject is in relax and conscious condition there is a higher possibility of involuntary head movement due to tremor, muscle contraction or relaxation involving to-and-from movement.

3.1.1 Human's head-neck system and unconstrained head motion

Human head-neck movement is extremely complex and is not completely understood. There is a considerable amount of literature published on this topic, however it is still impossible to model the head-neck system totally because of several factors namely; head-neck sheer complexity and lack of quantitative data on material properties, geometries and boundary conditions of individual tissues [Hannaford et al., 1986; Ferman et al., 1987; Winters, 1988; Hagemann et al., 1999; Moore et al., 2005]. The outcome suggests that there is tremendous variation in the types of biomechanical models to approximate the human head-neck system [Peterson and Richmond, 1988]. This is due to the consideration of task, for example, slow voluntary movement is differed than movement resulting from an impact to the head or during walking. In addition, researchers also suggest those horizontal and vertical movement kinematics is coupled as the same muscles are involved in both types of movement [Hannaford et al., 1986; Peterson and Richmond, 1988]. Typically the movement is assumed to be a straight path with a single peak and bell-shaped velocity profile [Plamondon, 1995; Plamondon et al., 2003; Cavanaugh et al., 2005].

However in this research, the problem is simplified from a task specification point of view in which force control is executed along the normal direction to the subject's head surface. Thus, it is necessary to establish the translational movement of the unconstrained head in relax and seated condition along this direction. A linear relationship between rotational and translational movement was established. An ellipsoidal shape was chosen to define the head geometry as it exhibits elevation as well as azimuth dependence [Duda et al., 1999]. To express the head geometry analytically, consider the head pitch movement (θ_p) around the rotation axis which is the centre of the ellipsoid as shown in Figure 3-11.

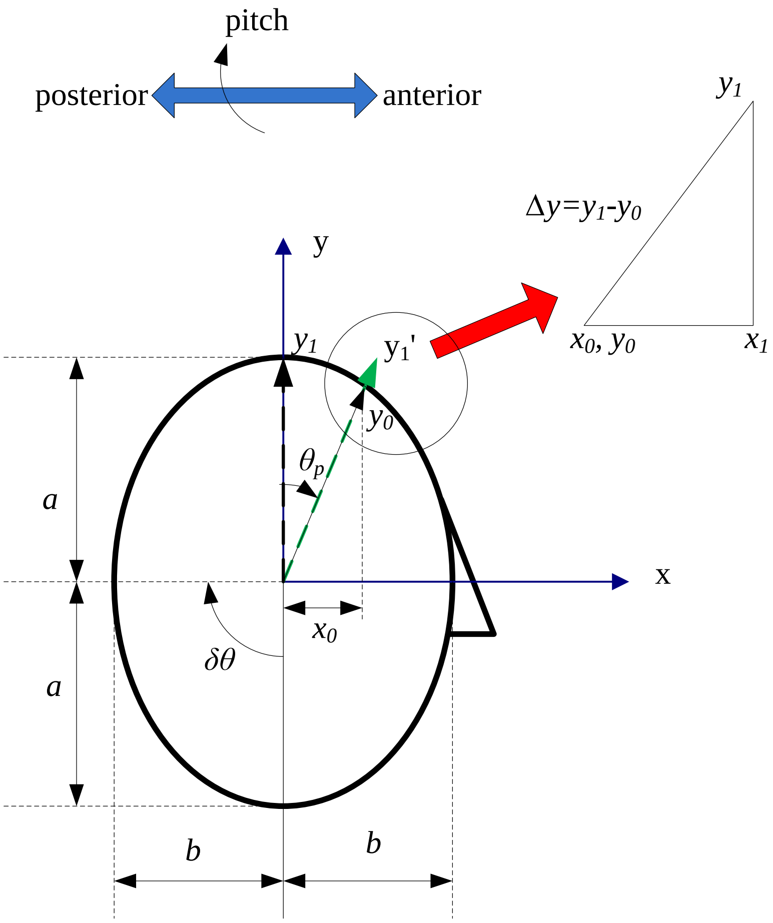


Figure 3-11 Ellipsoidal shaped head geometry *θp*: head pitch movement, *a*: distance from a centre rotation, *b*: half head width

As the head rotates about θ_p from start position y_1 to final position y_0, the translational distance can be expressed as $\Delta y = y_1 - y_0$. The following heuristic procedure is defined and yields a close approximation to determine Δy;

Find the point y_0,

$$y_0 = \frac{x_0}{tan\ \theta_p} \tag{3-1}$$

Find x_0 by using following ellipsoid equation;

$$\frac{x_0^2}{a^2} + \frac{y_0^2}{b^2} = 1 \tag{3-2}$$

Define the y_0 and gradient *m* as both y_1 and y_0 positions lie on the same straight line,

$$m = \frac{x_0}{y_0} = \frac{x_1}{y_1} = \frac{1}{tan\ \theta_p} \qquad (3\text{-}3)$$

Combining equations (3-1), (3-2) and (3-3), thus x_0

$$x_0 = \sqrt{\frac{a^2 b^2}{\left(\frac{a}{tan\ \theta_p}\right)^2 + b^2}} \qquad (3\text{-}4)$$

where, a is a distance from centre of rotation axis (approximately 30 mm inferior to the interaural axis [Moore et al., 2005]) to the y_1 point and b is the half of human head width. Based on Stephen [1996], Ferman et al. [1987] and Moore et al. [2005] works, a, b and θ_p can be defined as 160 mm, 95 mm and 10° respectively.

For further analysis, the head-neck kinematics [Peterson and Richmond, 1988] is related as following equation;

$$\frac{dx}{dt} = r.\frac{d\theta}{dt} \approx \vartheta = r.\omega \qquad (3\text{-}5)$$

where r, (x,θ) and (υ,ω) are the vector directed from the axis of rotation, translational and angular position and translational and angular velocity respectively. To guide this analysis, the head movement parameters were estimated by considering various research studies. Ferman et al. [1987] suggest that the subject head oscillate about 0.25 rad/s to 1.05 rad/s (amplitude about 10 degrees) in the horizontal, vertical or torsional plane during seated position while fixating target at optical infinity. This corresponds to 40 - 160mm/s translational velocities. This is almost consistent with the findings of Chan et al. [1996] that investigated the amplitude of head rotation for the subject seated stationary in defined ‘relax’ and ‘still’ position with the head in a resting and facing forward. Their studies found the approximately similar oscillations of 0.5 rad/s to 1.5 rad/s in both conditions. Other studies also suggest that the stabilization of the head ‘in space’ for the seated subject can be achieved at frequency approximately less than 1Hz [Gresty, 1987; Moore et al., 2005]. Subsequently this study will therefore considered and focus on motion along the normal direction of the surface environment within identified velocity movement. These calculation parameters can be considered as motion disturbances to the force controller and will be used in following simulation model and experimental programme chapters.

3.4 Force-controlled approach

To automate the conventional hand-held TMS procedure, the force-controlled system would involve the robot arm holding the coil, depending on the required treatment and subjects. The force-controlled approach is simplified as follows;

- A six axis force/torque sensor is integrated onto robot end-effector, giving robot compliance during tele-operation mode and sensing capability of the contact force during tracking mode. Details of the force/torque sensor system are given in Chapter 4.
- Tele-operation mode will enable the operator to directly control the robot arm by his/her own senses regarding to his/her experience and judgement. Thus, the operator is able to positioning the coil onto hot spot position more efficiently with the help of the compliant arm.
- Tracking mode is specifically designed to automatically maintain a low contact force of ≤ 5N between the coil and subject’s head. In this mode, the number of force control robot axis motion can be restricted to along normal direction to the subject’s head surface, thus avoiding the possibility of the robot to move off in unknown directions if a failure should occur.
- The robot arm can be categorized as an active device, thus the safety issue is always the most important concern. Several methods are proposed to enhance the safety of the procedure and will be discussed in Chapter 4.
- Besides contact force sensing capability, due to the task constraint, a suitable force control scheme is necessary to maintain a light contact force while tracking the subject’s head movement.

3.5 Summary

This chapter has presented an analysis of hand-held TMS procedure that leads to the idea to automate the procedure involving implementation of robot force control. The TMS task specifications have described the requirements of force-controlled TMS robotic system. Force-controlled TMS robotic system can be divided into two phases: Preoperative and intraoperative in which both of the phases will fully utilized robot force control feature. Two analyses were conducted in which defining a contact forces during conventional procedure and possibility head movement occurs during treatment. These analyses will provide a guide in further development of force control strategy for TMS robotic system. The following chapter will provide a description of the design of force-controlled TMS robotic system.

CHAPTER 4

DESIGN OF FORCE-CONTROLLED TMS ROBOTIC SYSTEM

Prior to the implementation of force controller to a physical robot, the hardware and software issues of the overall robot systems have to be addressed. Thus, the chapter focuses on the development of the six degree of freedom TMS system which consists of two main subsystems: the robot subsystem (manipulator arm and robot controller) and Force/Torque data acquisition system. In the initial phase of the work, the Force/Torque (F/T) data acquisition unit developed by Po-ngaen [Po-ngaen, 2006] was implemented. However, due to some inconsistency in data acquisition performance a new F/T system was acquired.

An overview of the hardware configuration of six degree of freedom TMS robotic system is provided in section 4.2. Section 4.3 provides a historical account of development of the F/T data acquisition system, tracing its origins from Po-ngaen F/T DAQ system to the current system. Section 4.4 deals with the software configuration and its implementation issues, focusing on synchronization problems that arise in real-time systems. The evaluation of the proposed system will be discussed in section 4.5. The remainder of this chapter is devoted to the TMS system safety system analysis and requirement, and finally provides the guideline for practical implementation of safety system.

4.1 Control System Design Criteria

As described in Chapter 1, the main design objective of the current research program was to develop and design a multi degree of freedom TMS robotic system which would ultimately improve current TMS procedure. The selection of Stäubli TX60 robot manipulator arm to handle TMS operation is described in detail in section 4.2. The control system for the TMS robotic system was designed to perform the following main functions:

- Monitor the positions of all six robot axes in real-time, relative to a chosen null position.
- Use the data from the Force/Torque transducer to calculate the ALTER path modification data for the robot manipulator arm axes.

The implementation of these requirements will be discussed in detail throughout the remainder of the chapter.

4.2 Hardware Configuration

The development of control architectures has a long tradition in the research community and is the foundation of successful research in robotics. Real-time capability is a key requirement in the design of robot control architecture as the events in control of the manipulator arm must be processed in the range of milliseconds. The hardware configuration presented is based on following computers;

- CS8C controller - that controls the robot via digital power amplifiers dedicated to each axis of the TX60 robot manipulator arm. The J205 Ethernet port with the IP address 192.168.0.254 (mask 255.255.255.0) is used to communicate with a QNX host PC.
- QNX host PC - with a clock speed of 2934MHz was used for real-time system control. Ethernet port with the IP address 192.168.0.252 was used to communicate with the CS8C controller.

Figure 4-1 shows a general view of the force-controlled TMS robotic system.

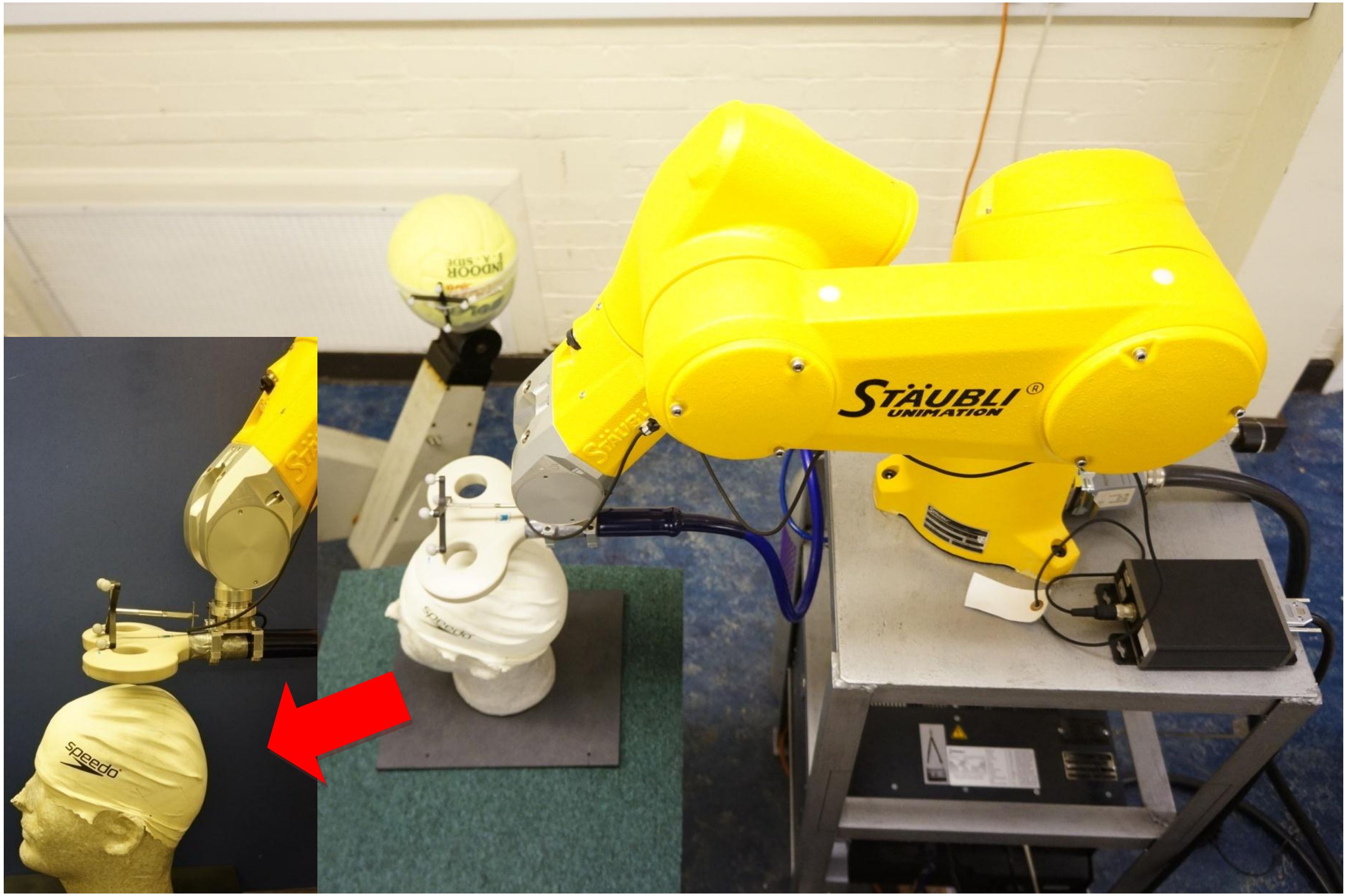


Figure 4-1 The overall TMS robotic system

Figure 4-2 illustrates a schematic diagram of proposed TMS robotic system and consists of all main components of the system and system control architecture. The six degree of freedom Stäubli TX60 robot manipulator arm is used to position the stimulating coil allowing both coil position and orientation to be controlled. Small head movements due to postural sway or the subject altering their sitting position can be detected and corrected in real-time using an optically based NDI Polaris Spectra 3D tracking system which tracks the head position and orientation using passive reflective markers [Yi, 2012]. A six-axis ATI Mini 40 force/torque transducer was mounted between the robot end-effector and the coil, to provide force feedback to maintain the coil in contact with the subject's head by a light force of not more than 5N. If the subject attempts to move towards the robot, the contact force will increase momentarily and the robot will then move away to return the force to the target level. By using a combined position and force feedback loop, the subject will be able to make small movements in any direction whilst the robot maintains the coil in the correct position and orientation.

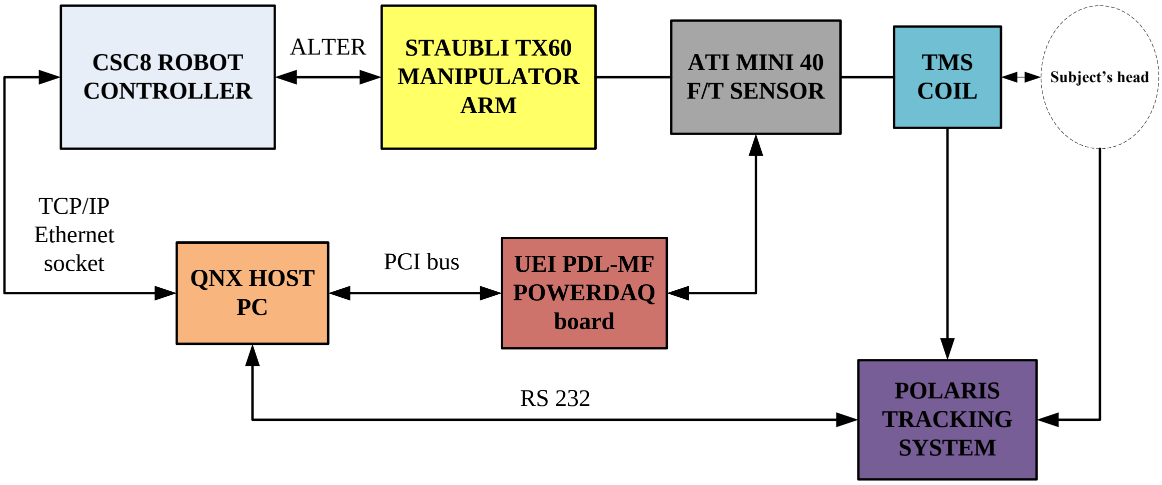


Figure 4-2 Schematic diagram of proposed TMS system

The TX60 robot manipulator has 9kg maximum load capacity and 600mm reach between axis 1 and 5. The robot specification is given in Appendix A. The TX60 robot incorporates 'ALTER' feature of VAL3 enables CS8C robot controller to control real-time path modification using position control relative to a chosen coordinate frame. The details of how this system was achieved and its effectiveness are discussed in section 4.4.

The six component ATI Mini40 force/torque sensor is installed between the robot end-effector and the coil. The sensor can measure up to 120N force and 2Nm torque with a resolution of 0.02N and 0.00025Nm respectively. The following section 4.3 will discusses further details of the F/T system.

The system is controlled by a single PC running under the QNX operating system and is dedicated to acquire data from the F/T DAQ system and Polaris tracking system and then performing necessary calculation on these data. The host PC is also used to initialize the main system components, initiates the real-time control system and handles real-time communication between the system components.

4.3 Development of Force/Torque Data Acquisition (DAQ) System

Prior to implementation of force control on TMS robotic system, the force and torque signals induced by the interaction between robot arm end-effector and subject's head have to be processed. An ATI Gamma F/T sensor was initially used to measure F/T signal at the beginning of the DAQ development. The six-axis sensor can measure up to ±130N force and ±10Nm torque with a resolution of 0.1N and 0.0025Nm respectively.

The F/T sensor consists of two main components; the transducer and the F/T controller. The transducer is a compact, rugged and monolithic structure of aircraft grade aluminium that converts force and torque into analog strain gauge signals. Appendix B provides the specification of transducer and the F/T sensor system controller unit. The transducer incorporates on-board signal conditioning and a multiplexer for the strain gauges and is designed for high overloads by using quality silicon strain gauges in combination with high precision pins. The analog strain gauge signals are transmitted to the controller unit through electrically shielded high-flex transducer cable. The six strain gauge output signals are biased and then resolved into their six Cartesian force/torque components by performing a calibration matrix calculation. The resolved data can also be transformed into any pre-selected tool frame and the controller unit outputs the data through parallel or serial port at a rate up to 300Hz. A standard set of system software commands offer the user a simple programming language which allows the user to set up communication, request data and monitor the sensor conditions.

To implement a real-time control system, it is important the F/T signals acquisition and processing time occurs at high sampling rate, and the ATI controller unit was replaced with a high speed control unit developed by Po-ngaen [2006] to increase the force data sampling rate from 80Hz to 1.4kHz. Figure 4-3 illustrates the architecture of this data acquisition system.

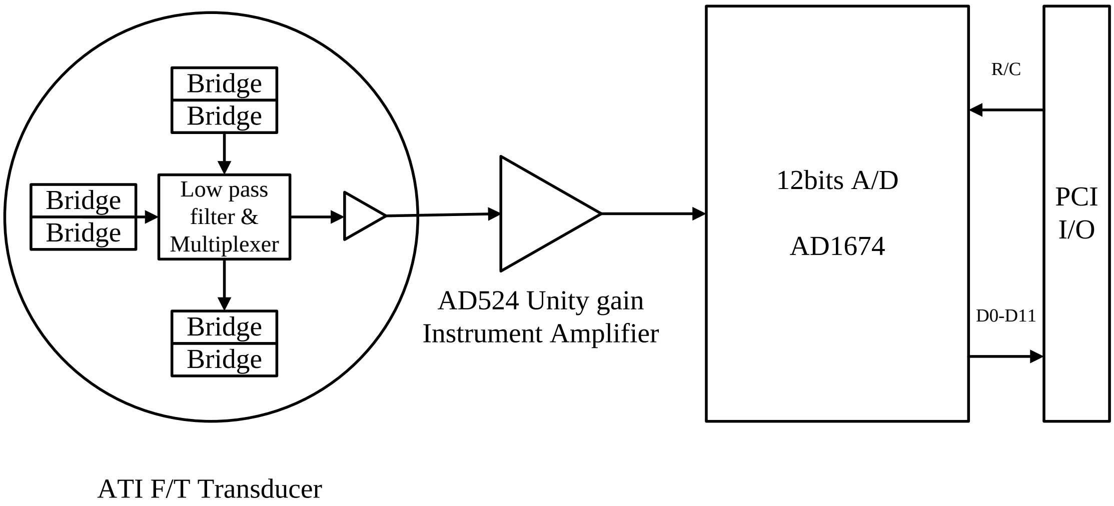


Figure 4-3 Po-ngaen’s Force/Torque Data Acquisition System [Po-ngaen, 2006]

Po-ngaen [2006] demonstrated the high performance of the DAQ system is achieved by retaining only the basic signal processing function to maximize the dynamic range of the analogue/digital hardware. Consequently the sensor controller task is simplified and involves amplification of the strain gauges signals using a precision instrumentation amplifier, then quantized using a 12-bit ADC converter before being latched over the PCI digital I/O card interface on the host PC. The software running on the host PC carries out the ADC conversion initialization and signal biasing by first recording the six channel outputs immediately after a reset. These values are then subtracted from all subsequent digital readings to effectively null the digital data. The six strain gauges signals are then converted to the six Cartesian force/torque digital data by using a 6 x 6 calibration matrix. The system can effectively improve system bandwidth with the acquisition and processing time of 0.7ms for each iteration process.

A simple test was conducted to evaluate the performance of Po-ngaen’s DAQ system and quality of the acquired F/T signals. The test description is discussed in the

Appendix E-1. The test results discard the DAQ system as the 12-bit ADC limits the force resolution to 0.5N. To overcome this limitation, a new Mini40 sensor with a new DAQ card has been proposed to replace the current F/T sensor and DAQ system.

4.3.1 Mini40 F/T transducer and PDL-MF board

A small six component DAQ F/T system (ATI Mini40 model) was chosen to supersede the ATI Gamma transducer in order to obtain a higher resolution force/torque data. Figure 4-4 shows the ATI Mini40 Force/Torque transducer.

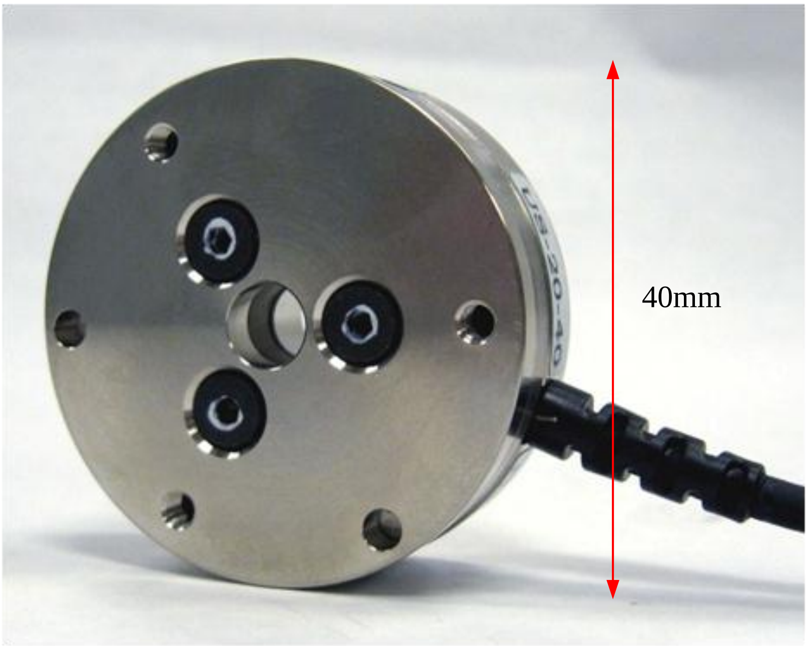


Figure 4-4 ATI Mini40 Force/Torque transducer

The 40mm diameter six-axis transducer with a height as 12mm can measure up to ±80N force and ±2Nm torque with a resolution of 0.02N and 0.00025Nm respectively. The Mini40 system consists of three components; the transducer, the interface power supply (IFPS) box and a PCI based data acquisition board. The transducer is light weight (only 50grams) and has built-in overload protection. The sensed forces and torques are converted into analog strain gauge signals and transmitted through an integrally attached transducer cable to the IFPS box. The IFPS box conditions the transducer signals and then passes them to a data acquisition board through an umbilical cable. The IFPS box also provides power to the transducer and electronics unit. Figure 4-5 illustrates the electronic hardware outline of component system. Essentially, the electronics hardware component measures the change in resistance of strain gauge and its accompanying software converts this change into useable force and torque components.

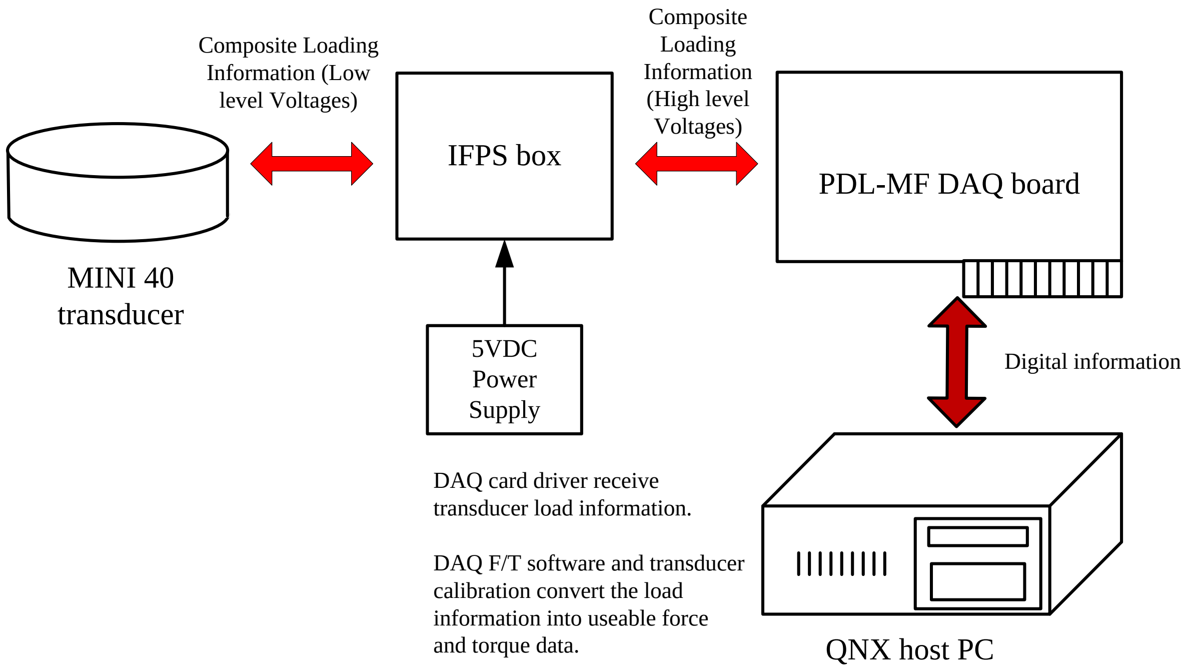

Composite Loading Information (Low level Voltages)
Composite Loading Information (High level Voltages)
IFPS box
PDL-MF DAQ board
MINI 40 transducer
5VDC Power Supply
Digital information
DAQ card driver receive transducer load information.
DAQ F/T software and transducer calibration convert the load information into useable force and torque data.
QNX host PC

Figure 4-5 Electronics Hardware Outline [ATI, 2010]

The transducer outputs are designed to work with a differential input to the DAQ system to achieve best performance. A 5VDC power supply is used to power the IFPS electronics and transducer. The transducer and DAQ card connection are illustrated in Appendix C-3. ATI recommends the F/T transducer uses a National Instruments (NI) M-series data acquisition board, however, this card is only supplied with a Windows device driver and the ATI F/T application software is Windows based. Thus, a Multifunction PowerDAQ PCI board (PDL-MF-50 Lab Series) manufactured by United Electronics Industries (UEI) was selected as it supports the QNX OS architecture and fulfils the DAQ F/T system requirements. The board uses a Motorola 56301, a 66MHz DSP that ensures a highly efficient interface with the PCI bus and also provides control over all the subsystems (specification given in Appendix C-4). The PowerDAQ application programming (API) which is integrated into the PowerDAQ dynamic link-library (DLL) allows the user to communicate with the board and includes a set of functions to perform the data acquisition process.

Six channels of the 16-bit Analog Input (AI) subsystem was used to acquire the analog strain gauge data from the transducer, and provides selection of input mode (single-ended or differential), polarity, gain settings, range settings, Channel List set up as well as trigger and clocking control. The DAQ board is designed to collect data at its maximum rated speed (50k samples/sec) and the DSP controls the acquisition process and stores the data locally. All digitized signals are transmitted into a 1k A/D FIFO before being transferred into the host PC memory. The host PC performs acquisition of the signals from the A/D FIFO through several available programming techniques. Initially, the Advanced Circular Buffer (ACB) mechanism was employed to run continuous acquisition as this method is appropriate for fast speed acquisition rate up to the maximum board speed. In using this method, the user works with one part of a buffer while the A/D FIFO fills the other half. The process runs continuously each time the board driver issues an event to notify the user application that data is ready for retrieval. The advantage of this mechanism is the data acquisition occurs at maximum board speed. However the main drawback is that the user application is only capable of receiving up to 10 events per second as the data is only available after the A/D FIFO becomes half full. Consequently this method will introduce a significant delay in real-time control loop application since the incoming data is in a ‘block’ format that contains multiple number of samples.

In view of this limitation, it was decided to implement a single scan mode in which the acquisition is initiated with a command by monitoring the external Channel List (CL) clock or Scan Clock. The CL clock instructs the control logic when to begin processing a full scan of all channels through Channel List. The maximum number of samples acquired is less than the minimal size of the A/D FIFO, so all the data remains in the FIFO and it can transfer the data without the need for ACB in host memory. The flowchart of data acquisition is illustrated in following Figure 4-6.

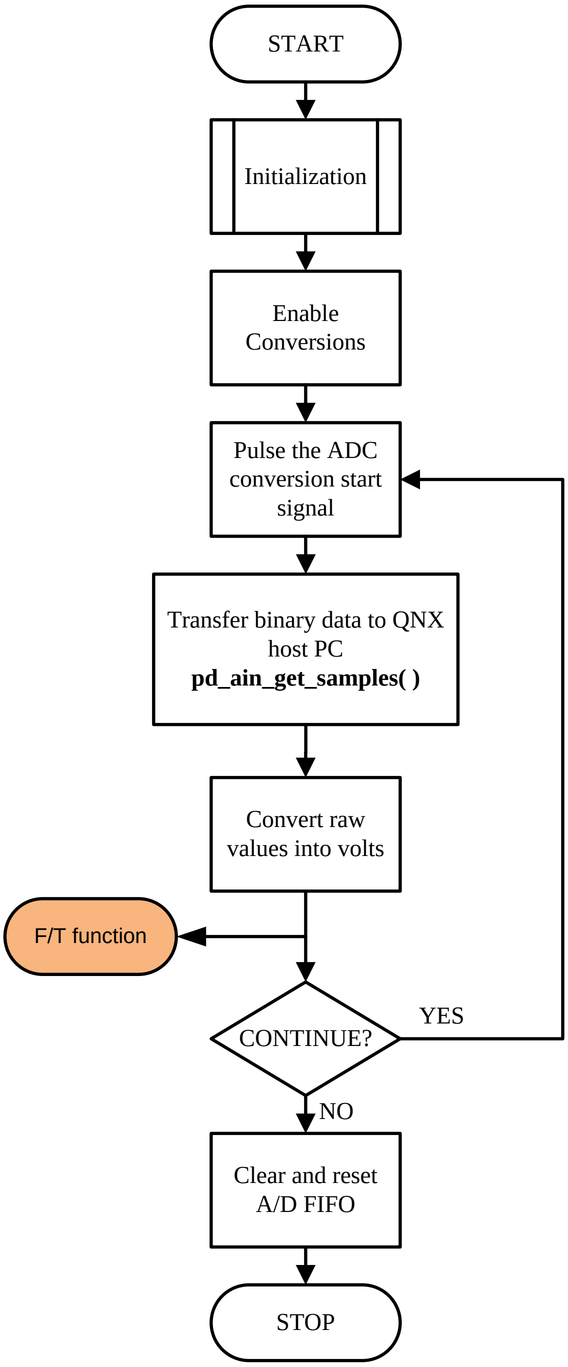


Figure 4-6 The flowchart of single scan mode data acquisition process

The process begins with an ‘Initialization’ routine in which the board is reset followed by a configuration of the input range mode, type of clocking and triggering as well as Channel List entries (number of channels). Next, the analog input conversion command is enabled to complete the AI configuration initial (to enable the data acquisition process) before triggering the AI start event to begin sample acquisition in software timed mode. The ADC conversion start signal is pulsed and the **pd_ain_get_samples( )** function is used to get the acquired samples from the A/D FIFO and move them into the array declared in the user application. To avoid an overrun condition when the numbers of scan equal to the A/D FIFO size (the subsystem can lose synchronization at this point) it is recommended that a call **pd_ain_clear_data( )** function is made after data transmission. The PowerDAQ API also encompasses the function to convert the raw data values to its equivalent voltage that is required by ATI F/T data conversion application program (Figure 4-7). The method can effectively acquire and process the six components of strain data at 500Hz sampling rate, which is sufficient for this application. A brief description of functions that were used in single scan data acquisition is described in Appendix C-5.

ATI provides an ANSI C library code for developing non-Windows based applications. The library provides programming interface language to load a calibration file into memory, configures the sensor system, applies tool transformation, bias function as well as convert the retrieved voltages signals from DAQ card into useable force and torque components. The flowchart of the F/T user application is illustrated in Figure 4-7.

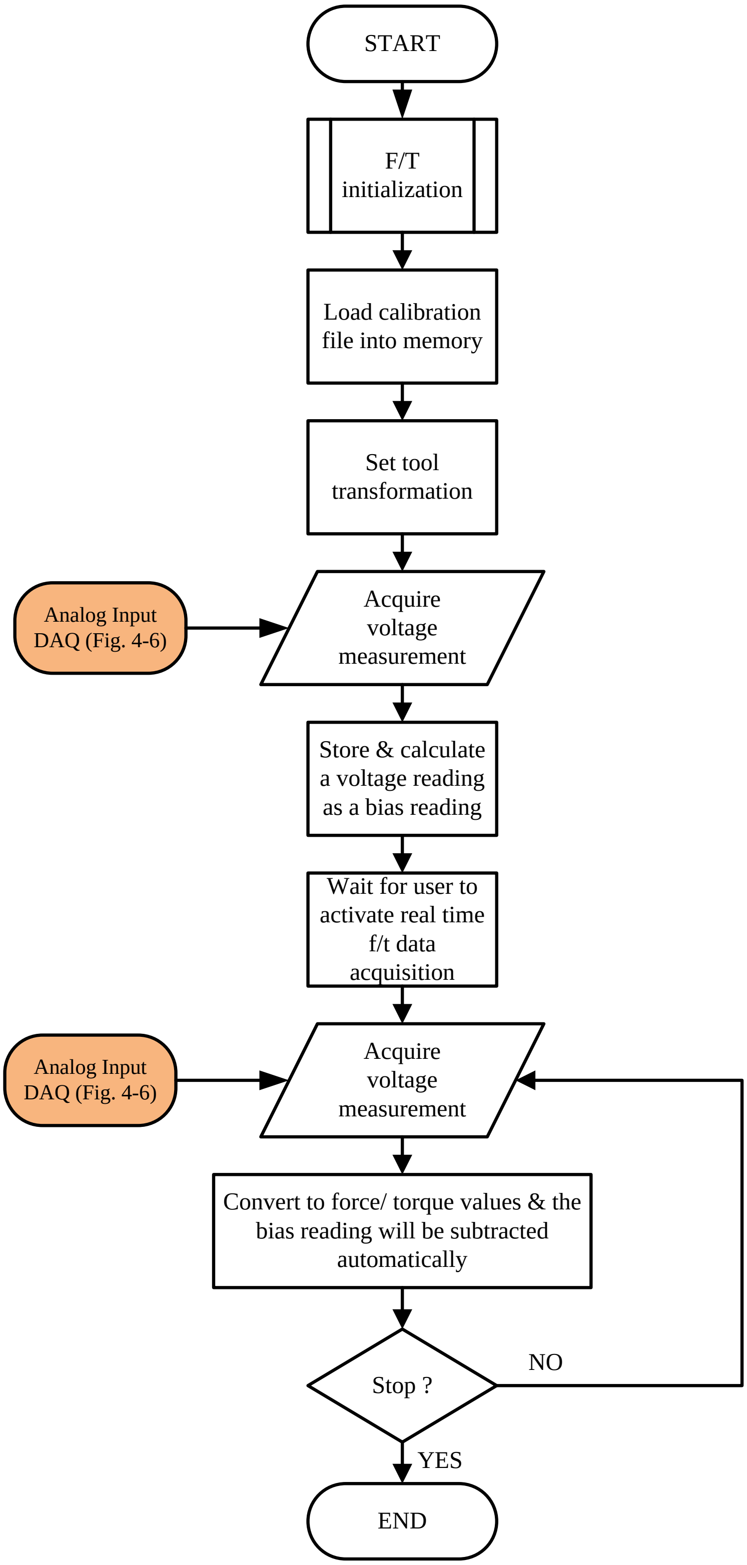

START
F/T initialization
Load calibration file into memory
Set tool transformation
Analog Input DAQ (Fig. 4-6)
Acquire voltage measurement
Store & calculate a voltage reading as a bias reading
Wait for user to activate real time f/t data acquisition
Analog Input DAQ (Fig. 4-6)
Acquire voltage measurement
Convert to force/ torque values & the bias reading will be subtracted automatically
Stop ?
NO
YES
END

Figure 4-7 F/T function software

The application begins with **createCalibration( )** function that is used to load a calibration file into memory. The calibration file includes the transducer stiffness matrix, force and torque units as well as coefficients used in real time calculations. The default calibration matrix resolves force and torque with respect to the transducer reference frame. The stiffness matrix is given in Appendix C-2. It is also necessary to perform a tool transformation to directly measure the forces and torques acting at an appropriate contact point on the coil. As can been seen in Figure 4-7, the device driver is activated to retrieve a voltage reading from the transducer and the **Bias( )** function is used for storing a voltage reading to zero the subsequent transducer output readings. This step is necessary to compensate the effect of gravity (tool weight) or other forces acting on the robot end-effector. Finally, the biased reading was converted to the resolved force and torque data.

4.4 Real-time external controller

A real-time system means that the timeliness of the calculation result is important [Stankovic, 1988; Gambier, 2004]. The TMS robotic system can be categorized as a hard real-time system as the time deadlines must be met, and any failing calculation result would be invalid. In general, building a real-time control system requires two stages; controller design and digital implementation. The controller design stage is normally defined by a control performance requirement while maintaining stability and rejecting any disturbances. The details about this will be discussed in Chapter 5 and 6.

The controller is executed cyclically according to the required sampling time; achieving the desired performance in which the control design is normally based on a time-discrete model which depends on the sampling period. In order to satisfy the requirement, it is necessary that the incoming data for the control signal is delivered as soon as possible. Hence, multiple control tasks should be scheduled to run at the implementation stage. All tasks must be scheduled with limited available computing resources as the chosen sampling time should consider the limited computation time provided by the hardware.

This section focuses mainly on the real-time operating system, robot communication, real-time path control using ALTER, synchronization approach and software process based structure which are key requirements in implementing the real-time control system.

4.4.1 QNX® Neutrino Real-time Operating System (RTOS)

The QNX® Neutrino RTOS is a true microkernel OS that enables the user to create a highly optimized, robust and reliable system. QNX RTOS technology is used in mission-critical applications such as medical instruments, process control applications, air traffic systems and Internet routers.

This research adopts the QNX Neutrino real-time operating system (RTOS) v6.4.0 which supports implementation of multi-tasking system for the system. QNX is based on microkernel architecture which runs every driver, application, protocol stack and file systems outside the kernel and provides safe memory protected user space. If any of these components fail, it can be automatically restarted without affecting other components or the kernel. The QNX OS was selected primarily because it satisfies the real-time requirement as well as reducing the system real-time complexity, thus ensuring the synchronization between the system control tasks.

Figure 4-8 illustrates the overview of QNX Neutrino microkernel architecture. The fundamental services such as signals, timers and schedulers are included in the microkernel. All other components (filesystems, drivers, protocol stacks, and application) and processes are isolated in memory-protected user space which communicates through a single and well-defined messaging layer. The layer permits the addition of plug-in and removal of software components at any time.

There are many considerations to be addressed when designing a multi-rate robotic system which consists of the host PC, the robot controller, the F/T system and Polaris tracking system that run at different control/sampling rates. The host PC must control these multiple hardware for concurrent process. Depending on the data flow in a control system, synchronization can be achieved in several ways. (See section 4.4.4).

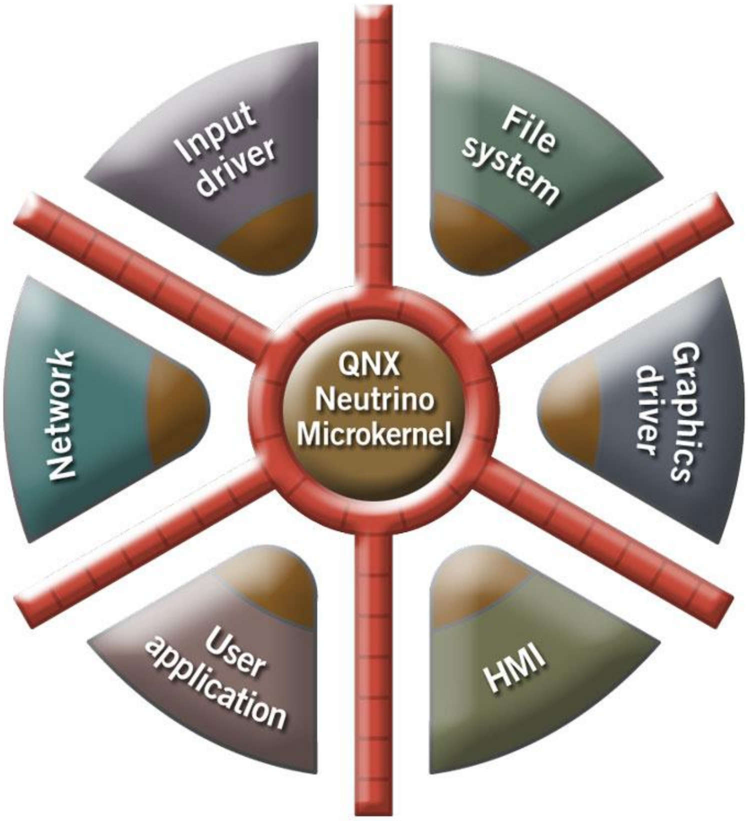


Figure 4-8 QNX Neutrino RTOS microkernel [QNX, 2004-2012]

4.4.2 Ethernet Socket Communications

The socket interface was initially developed for the Berkeley Software Distribution (BSD) variation of the UNIX operating system. The most widely utilized Transmission Control Protocol/Internet Protocol (TCP/IP) application programming interface (API) is used to communicate two application programs running on different computing systems and is based on a client-server relationship. The two common types of sockets that support communication in the TCP/IP environment are TCP stream sockets and User Datagram Protocol (UDP) datagram sockets. The TCP stream sockets provide a reliable byte-stream that supports the end-to-end form of data transfer. The UDP datagram sockets provide an unreliable datagram service where huge individual user messages (up to 65,500 bytes in length) can be sent to another socket. In our application, the TCP stream socket has been chosen to provide communication between CS8C robot controller and the QNX host PC with a fast transfer speed (100Mbit/s) and error checking mechanism which guarantees packet delivery. No additional I/O is needed as the CS8C robot controller and standard computer support this type of communication.

The QNX Neutrino RTOS v6.4.0 host PC and CS8C robot controller communicate with each other via an Ethernet socket communication. A CAT5 cable is used as physical communication between the host PC and robot controller based on the Ethernet network IEEE 802.3 standard. The system uses a server-client TCP Ethernet communication protocol where the robot controller (server) listens for a request from host PC (client) to establish a connection. Once the connection is established both the server and client can transmit and receive data for the system operation.

As real-time execution is an essential requirement, the time axes of the robot and the sensor need to be synchronized. A per-process timer routine was developed to perform data transfer for the TCP socket protocol on the QNX host PC which has to be synchronized such that the CS8C robot controller has sufficient time to acquire position commands every 4ms. Furthermore, the host PC also has to acquire and compute calculation of force/torque data and update position data to the robot controller within the allotted per-process time using a scheduling thread and timer. Communication between the host PC and the robot controller is only invoked when the Ethernet socket connection is activated on the robot controller using the VAL3 instruction called **sioGet( )**.

Once the position data is received in the input read buffer, the robot controller parses the string format data to the desired robot geometric transformation type. The robot will immediately begin to move to the new desired position according to this data. Two programs have been developed based on VAL3 high-level programming language to handle communication and data parsing, and to control robot movement. These two programs are synchronized every 4ms for one cycle since the robot controller only supports specified time rounded down to a multiple 4ms which is CS8C robot controller update time. If the execution of a VAL3 synchronous task takes longer than the specified period, the current cycle ends normally and the next cycle is cancelled, causing an overrun error.

4.4.3 ALTER: Real-time control on a path

In principle, the system controls the robot movements in a similar way that a driver drives a car. If the trajectory is well known in advance, the system can optimize the speed of robot movement. This explains why the system does not wait for the current robot movement to be completed before taking the VAL3 instructions for the next movement into account. For better explanation, consider the following program lines;

1. Movej (pA, tTool, mDesc)
2. Movej (pB, tTool, mDesc)
3. Movej (pC, tTool, mDesc)

When the first instruction is executed, the robot starts to move towards point pA and the program then immediately proceeds to the second line, before the robot reaches point pA. The system records that after pB the robot must move to pC and probably the robot is still moving towards pA. This successive robot movement cannot be interrupted until the robot completes its movement. Consequently this will be problematic in the TMS robotic system implementation since the robot is required to change its trajectory in real-time based on the force/torque sensor information. For this purpose, the ALTER protocol (available as an option to the existing standard VAL3 language) is adopted to provide real-time robot path modification.

Stäubli offers the *Cartesian* ALTER feature to modify the nominal robot path using an external sensor and this feature is immediately effective during robot movement. The feature is implemented in TMS robotic system due to its ease of application since no hardware modification was required and the complicated task of forward and inverse kinematic transformation for the robot are performed within the VAL3 controller. The ALTER mode of VAL3 is invoked using the **alterBegin( )** instruction within a VAL3 program and it continues until the **alterEnd( )** command is executed or the program aborts for some other reason. In the latter case, this may be due to the data error from QNX host PC, or the transmitted path modification data would result in excessive axis speed or a robot position which is out of range.

A VAL3 program is written to prescribe the position and orientation in which the robot is commanded to move to, whilst the ALTER feature provides an outlet for VAL3 to integrate external sensory input to modify the robot path in real-time. In the TMS robot application, the VAL3 program instructs the robot to stay where it is (pre-defined start position), while the ALTER data modifies the robot's position with respect to its start position as described by equation (4-1). The transformation can be interpreted as the deviation referred to initial position.

$$P(t) = P_d + \sum_t \delta P(t) \tag{4-1}$$

where $P(t)$is the desired ALTER position, P_d is the initial robot position when ALTER mode is activated and $\sum_t \delta P(t)$ is a sum of elementary position increments. The tool centre point (TCP) is set as ALTER reference frame since the deviation of the robot movement is measured relative to the TCP which is the centre of the figure-of-eight shaped coil. The velocity limits for the ALTER movement are defined by the **alterBegin( )** instruction. In TMS robotic system the **alter***(dx, dy, dz, rx, ry, rz*) command is applied every 4ms where the robot translates about *dx*, *dy* and *dz* along *x*, *y* and *z*-axes and rotates *rx*, *ry* and *rz* about the *x*, *y* and *z* axes. This must be computed carefully so that the resulting arm position and speed commands remain smooth and continuous in order to achieve the desired quality of robot arm motion and behaviour.

4.4.4 Synchronization procedure

The system's real-time capability lies in the fact that the complete robot joint movement is done at the same time or simultaneously with the sensor measurement. This involves multiple asynchronous processes which are run at different rates that must be synchronized. The true real-time OS design of QNX Neutrino ensures that time-critical tasks are dealt with by the right thread or process at the right time which keeps all the processes running as real-time application. Thus, offers user predictable system response times. Synchronization between the robot controller and the host PC is accomplished using two written program applications;

1) QNX program (host PC)

The application program consists of a several processes that handle F/T data acquisition (Task 1), force control law calculation and Ethernet socket communication (Task 2) as shown in Figure 4-9. A thread is created in order to run Task 1 concurrently and independently relying on DAQ board rate (in our case 500Hz). Task 2 which contains the force controller and Ethernet socket communication algorithms is a periodic timer scheduling process that triggers the real-time clock to generate periodic interrupts for the kernel. The routine is assigned to run in such desired time Δt_f, as a result the process can be synchronized with the robot controller processor update time.

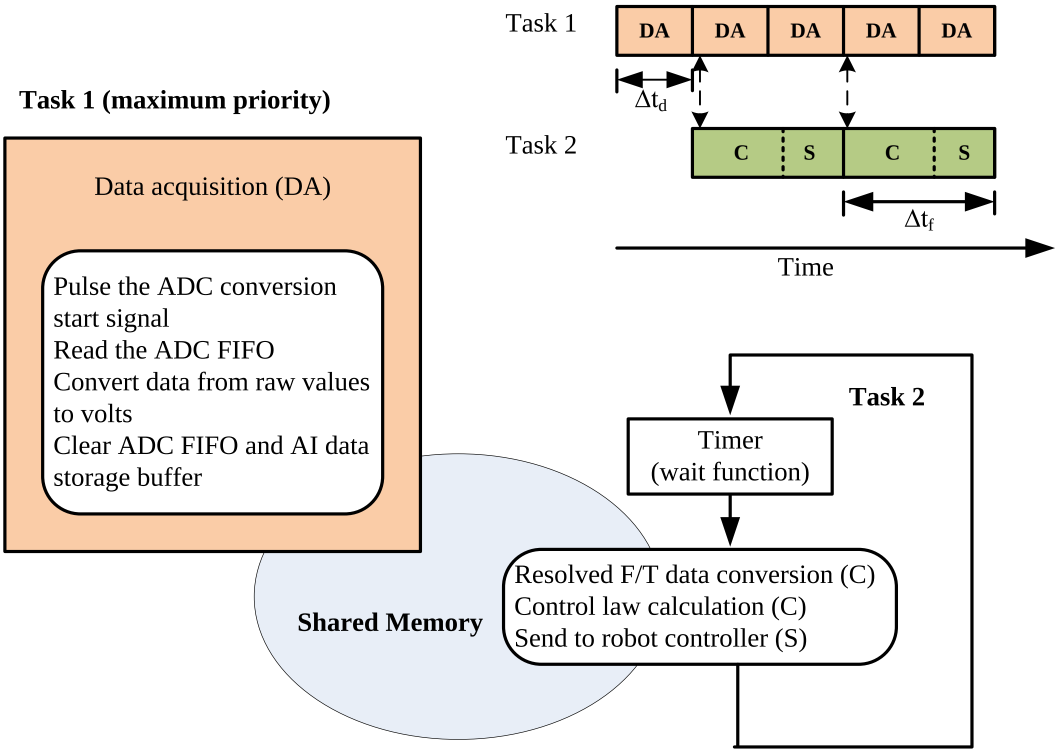

Task 1
DA
DA
DA
DA
DA
Δt_d
Task 2
C
S
C
S
Δt_f
Time
Task 1 (maximum priority)
Data acquisition (DA)
Pulse the ADC conversion start signal
Read the ADC FIFO
Convert data from raw values to volts
Clear ADC FIFO and AI data storage buffer
Shared Memory
Task 2
Timer (wait function)
Resolved F/T data conversion (C)
Control law calculation (C)
Send to robot controller (S)

Figure 4-9 Real-time application program to handle F/T DAQ, control law and Ethernet communication tasks

2) VAL3 program (CSC8 controller)

Two task programs are executed to handle the received position data (communication and data acquisition task) as well as controlling the robot joint movement (ALTER motion task). As the robot controller system has only one processor (the processor controls the robot via digital power amplifiers dedicated to each axis of the arm), it can only execute one task at a time. Normally, simultaneous execution is simulated by very fast sequencing of the tasks that execute a few instructions in turn before the system moves on to the next task. The process can be described as asynchronous, as tasks are scheduled by the system so that they can be executed as fast as possible.

The robot controller sends the position and velocity command to the amplifier every 4ms; thus the ALTER task must be synchronized with this communication period so that the alteration speed remains under control. This can be done using a synchronous VAL3 task instruction. The syntax of the synchronous task instruction is as follows;

void taskCreateSync <string sName>, <num nPeriod>, <bool& bOverrun>, program(...)

where the program(....) is the specified task's name and <num nPeriod> identify the interrupt frequency for the task to execute. The system only support the time that is rounded down to a multiple of 4ms which is robot controller computational update time for each synchronized task.

One of the major problems encountered during the development of the TMS robotic system was finding a means to synchronize the two programs as the timings imposed by VAL3 present a number of problems associated mainly with system stability and bandwidth. Initially it was decided to run both of the VAL3 program tasks as synchronous tasks where both are scheduled to run at regular periods of time. These tasks are executed in sequence in the order in which they were created. A diagram showing the overall time required to compute both tasks as illustrates in Figure 4-10.

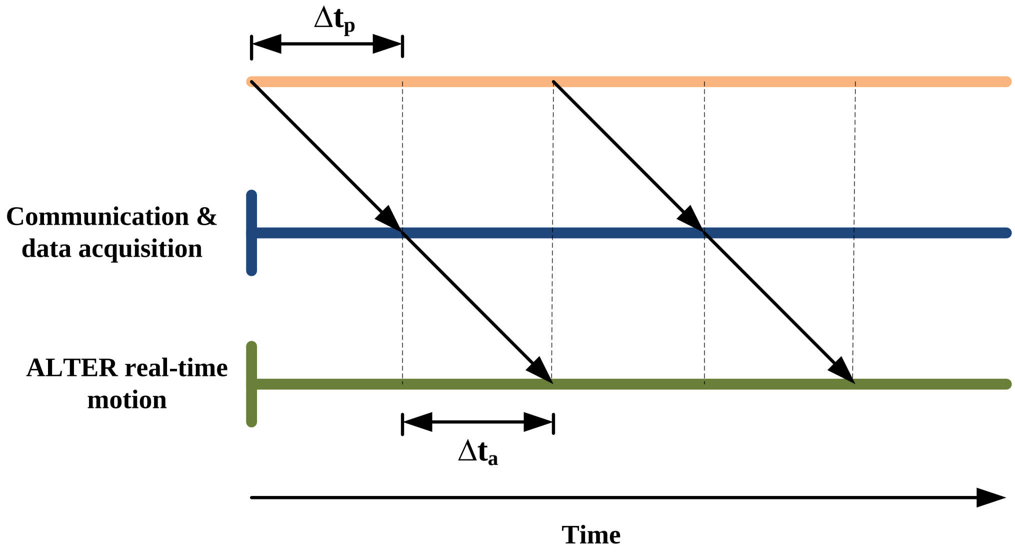


Figure 4-10 Timing diagram for both synchronous tasks

Δt_p is the time required to handle the communication and data acquisition task and Δt_a is the time required by ALTER to move the robot, thus the total overall cycle time ΔT would be:

$$\Delta T = \Delta t_p + \Delta t_a \tag{4-2}$$

Thus, the received data is ready after 4ms yet it takes another 4ms for robot to arrive at its commanded position.

Note that to synchronize all tasks on both the QNX host PC and the CS8C robot controller, the execution time of task 2, Δt_f (refer to Figure 4-9) must be set equal to the overall cycle time, ΔT. This was attempted but abandoned primarily because of the consequences of wasting considerable processor time performing both tasks in sequence, for which the overall cycle time is more than double the CS8C robot controller update time which inevitably lowers the bandwidth of the system. Another point of concern was with the **sioGet( )** instruction as a synchronous task. If there are any characters remaining in the input read buffer, **sioGet( )** fetches the data quickly, however if the input buffer is empty, the **sioGet( )** instruction will wait for the time-out delay before returning. This will also close the Ethernet socket connection. The system attempts to automatically connect the next time **sioGet( )** is used, but this is a lengthy and unpredictable process and can take several seconds to execute, which jeopardizes the system performance.

In view of these limitations, it was decided to develop an interrupt-driven technique establishing the communication and data acquisition task as an asynchronous task in which the task is scheduled by the system so that it can execute as fast as possible. The ALTER motion is set as a synchronous task which interrupts the current asynchronous task at periodic time, Δt_a as illustrated by following Figure 4-11.

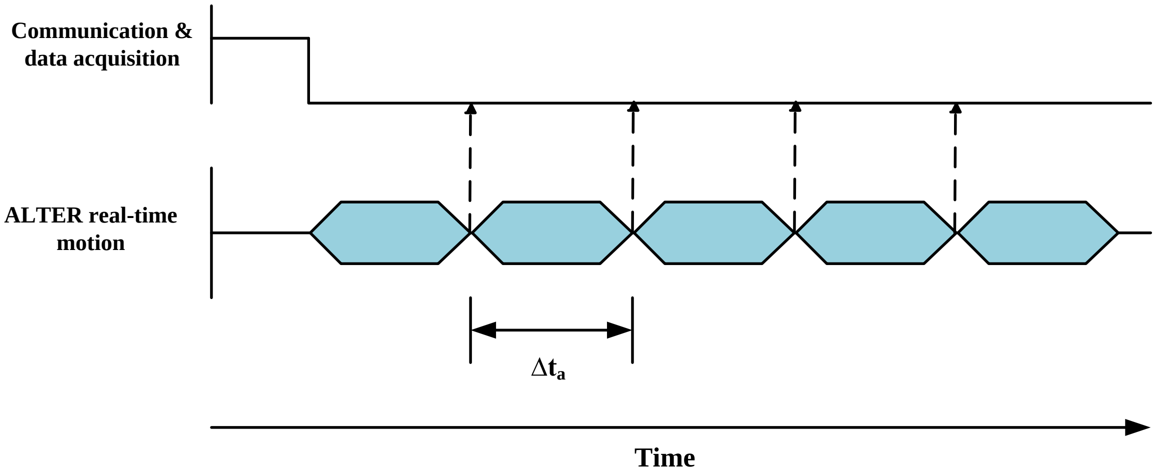


Figure 4-11 Interrupt-driven method

It can be seen that the time taken to execute one cycle control loop is reduced to 4ms. The former method was undesirable due to the length of time needed to performing all required tasks. The latter method is more a favourable solution as it can achieve the minimum computation cycle time of 4ms which subsequently improves the bandwidth requirement of the real-time system.

4.4.5 Process Based Structure

Contact force between the stimulating coil and the subject's head is measured using an ATI Mini40 six-axis force/torque sensor mounted on robot end-effector. The force control loop uses a control law to compensate the difference of force error between the desired and sensed force F_e and then transmits an incremental position and orientation x_d to the robot controller. The set positions and orientations x_d specify the desired motion of manipulator arm. The ALTER protocol in the VAL3 robot controller modifies the robot movement path in real-time. As discussed previously, due to the complexity of using a single processor to sequentially control all the various peripherals devices and the robot trajectory would result in low bandwidth or high latency system. To address these issues a multi-tasking concept was employed, in which each process functions as a dedicated process.

Figure 4-12 illustrates a flowchart of the inter-processing between the CS8C robot controller (server) and host PC (client). Both server and client initialization processes are executed only at the program start-up where the host PC monitors the force/torque data acquisition PCI board, whilst the robot controller establishes the Ethernet socket connection. Data from the force/torque sensor is processed by two main routines called *force/torque data acquisition* and *force controller*; both written in C language and compiled with GNU C compiler (gcc). As can be seen in Figure 4-12, *force controller* computes the resultant force/torque vector to the incremental position (and orientation) and is then transmitted to the robot controller via TCP/IP socket connection to modify the robot position in real- time such that it will maintain light contact force with the subject's head. The position demand is subsequently sent to the robot controller which servos the manipulator arm to the desired position and control the arm with the desired manner of simultaneous force and position control.

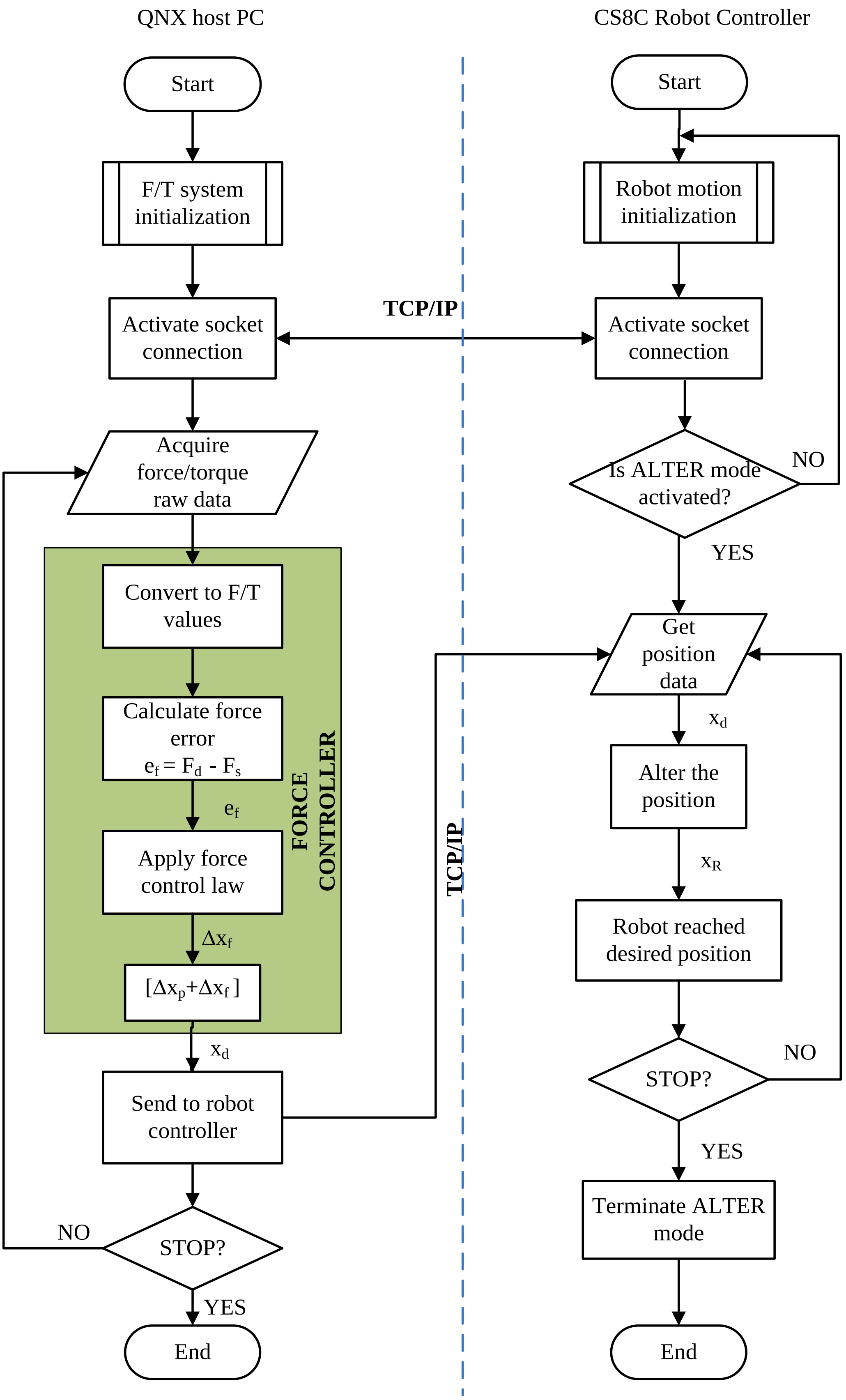


Figure 4-12 Communication structure between QNX host PC and CS8C robot controller

To establish a real-time network connection, a function named *TCP client socket* was written to connect the host PC (192.168.0.252) and the robot controller at specified IP address (192.168.0.254) and port (2300). Setting up the TCP client requires four basic steps, as illustrated in Figure 4-13.

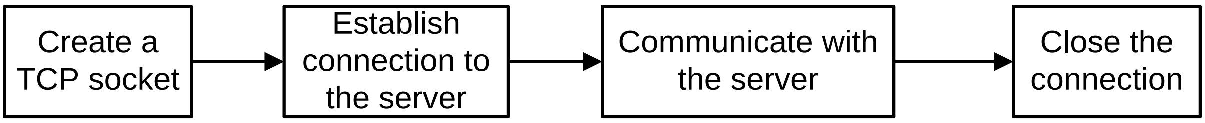

Create a TCP socket
Establish connection to the server
Communicate with the server
Close the connection

Figure 4-13 TCP/IP socket communication protocol

The function opens a TCP socket stream connection between the host PC and robot controller to allow both to exchange data using the **send( )** and **receive( )** commands via this open socket connection until the connection is closed. A TCP/IP server socket is created on the robot controller and the timeout is set to zero so the server is always in ‘listen’ mode with the client. In addition, either the **sioSet( )** or **sioGet( )** VAL3 commands are used to make a connection if there is no active connection. Once the robotic system network is established, both host PC and robot controller perform parallel computation to convert the resultant force/torque data to incremental position demand and then transmitted to robot controller to drive the robot arm.

Two VAL3 programming tasks are used to handle received position data, control robot movement and modify the robot nominal path, as shown in the following Figure 4-14. To begin with, the robot is commanded to move to the pre-defined target position until the force threshold is sensed. This corresponds to the preoperative phase (guarded move) of TMS procedure in which all robot axes is set up to position control as described in Chapter 3.

Once the robot is in contact with the environment, with the aid of the system graphical user interface, the operator activates the autonomous force-controlled tracking mode. The process is continuously operated until the procedure is aborted by the operator if the treatment is finished or if the system senses any errors.

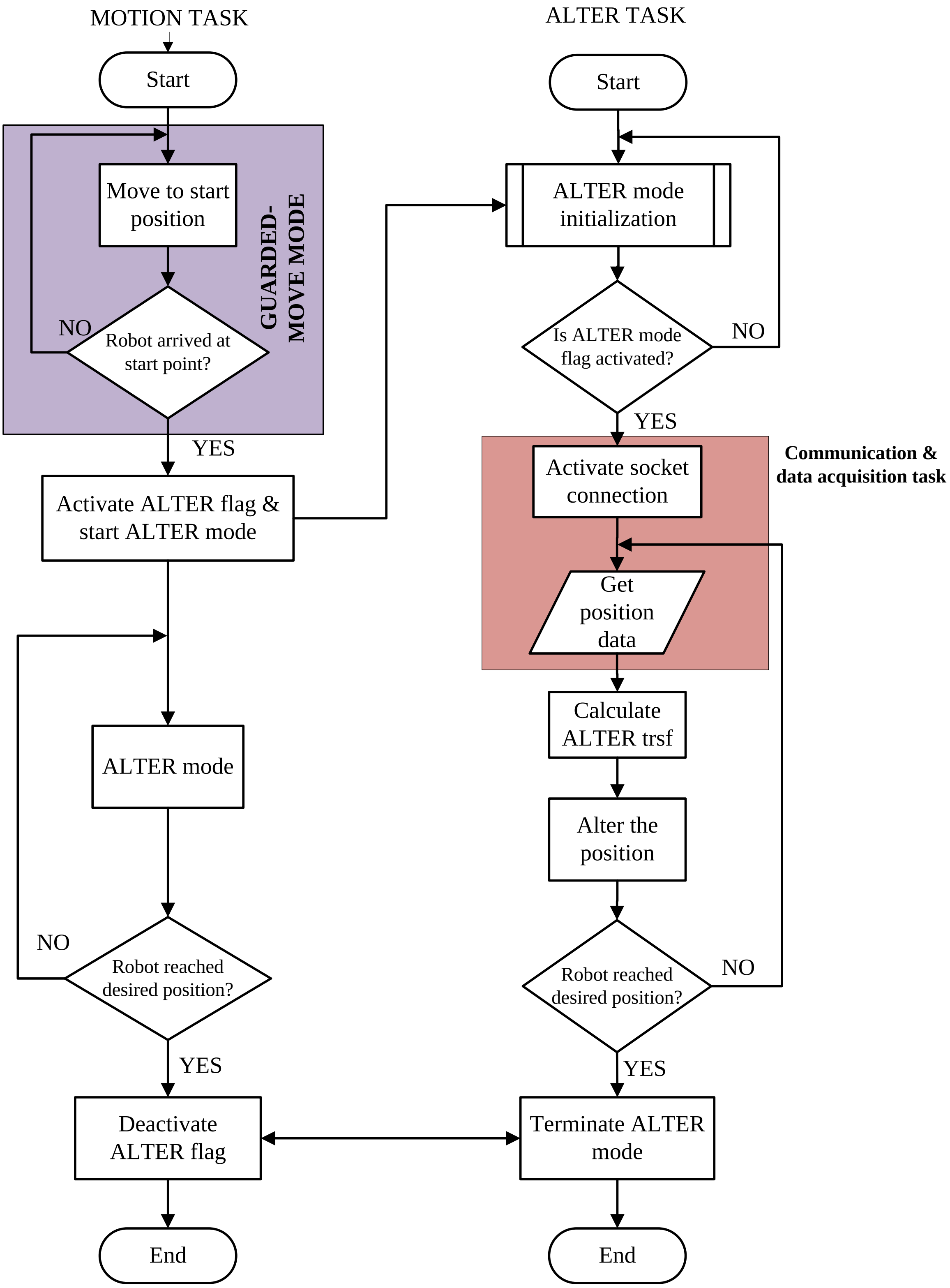

MOTION TASK
Start
Move to start position
GUARDED-MOVE MODE
NO
Robot arrived at start point?
YES
Activate ALTER flag & start ALTER mode
ALTER mode
NO
Robot reached desired position?
YES
Deactivate ALTER flag
End
ALTER TASK
Start
ALTER mode initialization
Is ALTER mode flag activated?
NO
YES
Communication & data acquisition task
Activate socket connection
Get position data
Calculate ALTER trsf
Alter the position
Robot reached desired position?
NO
YES
Terminate ALTER mode
End

Figure 4-14 Two main VAL3 program flowchart

All variables and flags are initialized in a separate initialization task. A VAL3 application always contains the default **start( )** and **stop( )** programs. All synchronous tasks are invoked and called in **start( )** program. The ALTER program must be synchronized within a 4ms communication period so that the ALTER speed remains under control as illustrated in the Figure 4-15.

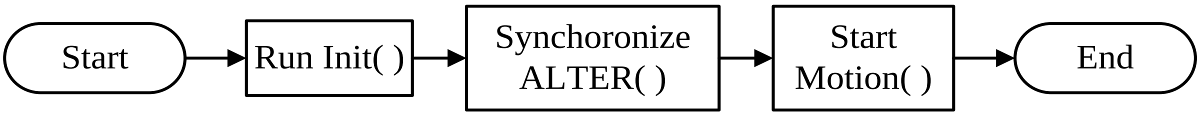


Figure 4-15 **Start( )** program flowchart

Tests were carried out in order to confirm the reliability of the socket communication between host PC and robot controller via socket TCP protocol at 100Mbit/s, as discussed in detail in appendix E-3.

4.5 Evaluation of the developed system

4.5.1 Mini40 Force/Torque DAQ system evaluation

The new Mini40 DAQ system was evaluated in a manner which would allow testing for reliability of the force sensing system. Several test scenarios were designed to determine its accuracy and to investigate the influence of robot dynamics (such as robot arm vibration and position configuration) on force/torque DAQ system output data. Force and torque data was recorded in a number of useful test scenario/modes namely: (1) Robot controller off; (2) Controller on, arm power off; (3) Controller on, arm power on; (4) Robot in ALTER mode (stationary) and (5) Robot in ALTER mode (moving in *x*, *y* and *z* direction).

In all modes, the Mini40 F/T transducer is mounted on the robot end-effector. The important aspect of the tests is that the robot arm configuration is located at the same position and orientation for each different repeated test to minimize any errors that may arise. For mode (1) to (4) each test was conducted for 30 minutes. To investigate the effect of robot movement to the sensing system, the robot is commanded to follow a 100mm polynomial trajectory for each axis at robot low speed for mode (5). Each mode was repeated five times in randomly sequence and the signal's error mean and standard deviation of force/torque output readings for all axes was quantified in Table 4-1 and graphically represented in Figure 4-16 to allow comparison between each mode. The signal's mean, μ calculation indicates the best estimate of the signal data and the standard deviation σ is a measure of how far the measured signal fluctuates from the mean value.

In Figure 4-16, it can be observed that there was no significant difference of the output of force signals for *x* any *y* axes which fluctuate around ±0.02 N from the mean value between the different modes. However, the standard deviation error indicates that for *z* axis mean force signal is approximately double (±0.04 N) than the other two axes value. This is expected as it is consistent with the resolution suggest by ATI company [ATI, 2010]. As can be seen from the Table 4-1, the higher resolution of F/T signal readings is obtained in which approximately 90% improvement compared to Po-ngaen system [Po-ngaen, 2006], indicates a precise measurement is achieved using a new DAQ system. There was also very little difference can be observed between the entire torque data signal for each axis.

To confirm effect of the robot ALTER movement, a series of similar tests were repeated in which the robot is commanded to move in three different directions. After examining the results, it is interesting to note that system actually performs better in all Cartesian directions. The comparatively smaller value in standard deviation over mean absolute error suggests that the system force response is less influenced by the robot movement. The result is significant as it proves an effective output reading with consistently similar range readings at all times. Based on this preliminary tests, it can be concluded that the new DAQ card can be used in place of the Gamma F/T controller and Po-ngaen DAQ system as the system provides a more reliable measurement with higher resolution performance.

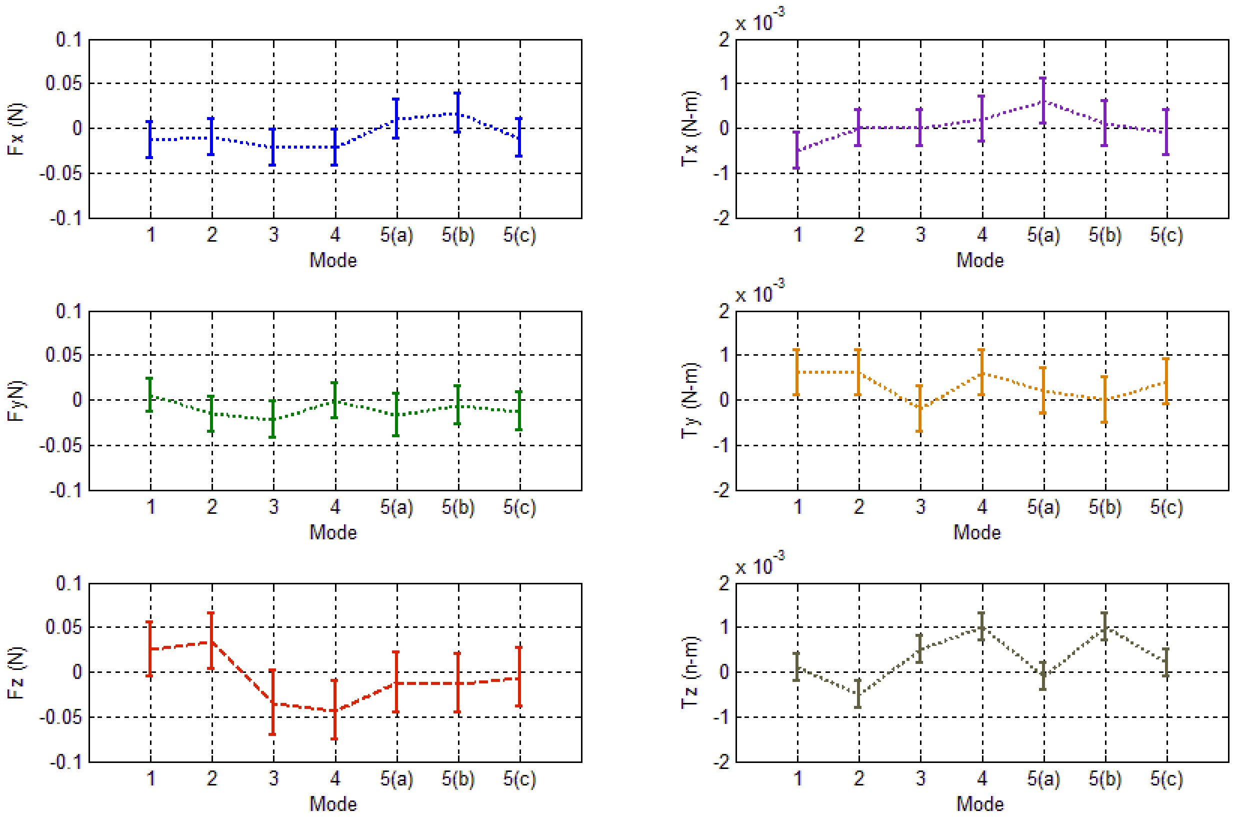
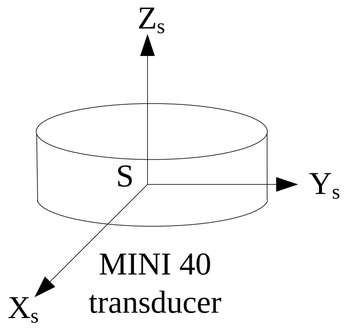


Figure 4-16 Mean and standard deviation of force/torque output readings

Table 4-1 Mean and standard deviation of force/torque output readings

Axis	x (N)	y (N)	z (N)	Rx (N-m)	Ry (N-m)	Rz (N-m)
MODE (1) Robot controller and arm power off						
Mean	-0.0134	0.0047	0.0251	-0.0005	0.0006	0.0001
Standard deviation	0.0195	0.0183	0.0306	0.0004	0.0005	0.0003
MODE (2) Robot controller power on and arm power off						
Mean	-0.0101	-0.0155	0.0340	-0.0000	0.0006	-0.0005
Standard Deviation	0.0195	0.0192	0.0309	0.0004	0.0005	0.0003
MODE (3) Robot controller power on and arm power on						
Mean	-0.0221	-0.0217	-0.0348	0.0000	-0.0002	0.0005
Standard Deviation	0.0204	0.0204	0.0357	0.0004	0.0005	0.0003
MODE (4) Alter mode (stationary)						
Mean	-0.0219	-0.00155	-0.0430	0.0002	0.0006	0.0001
Standard Deviation	0.0206	0.0194	0.0325	0.0005	0.0005	0.0003
MODE (5-a) Alter mode (x direction)						
Mean	0.0100	-0.0168	-0.0113	0.0006	0.0002	-0.0001
Standard Deviation	0.0222	0.0237	0.0336	0.0005	0.0005	0.0003
MODE (5-b) Alter mode (y direction)						
Mean	0.0168	-0.0060	0.0127	0.0001	0.0000	0.0001
Standard Deviation	0.0211	0.0205	0.0321	0.0005	0.0005	0.0003
MODE (5-c) Alter mode (z direction)						
Mean	-0.0114	-0.0132	-0.0642	-0.0001	0.0004	0.0002
Standard Deviation	0.0210	0.0211	0.0328	0.0005	0.0005	0.0003



4.5.2 Evaluation of ALTER real-time robot position control

A polynomial function was chosen to provide smooth trajectories for the robot as well as to simulate the head movement. Using polynomial, the motion profile (position, velocity and acceleration) for the robot real-time execution can be defined. For straight-line movements between two points it is desirable to have a velocity and acceleration that are zero at the beginning and end of the movement. A polynomial with odd power is needed to ensure the required motion profile, maintain the continuity of position, velocity and acceleration and results a minimum jerk movement. Flash and Hogan [1985] proposed the following expression (5th order polynomial) to obtain smooth unconstrained point to point movements;

$$x(t) = x_0 + \left(x_0 - x_f\right)(15\tau^4 - 6\tau^5 - 10\tau^3) \tag{4-3}$$

where $\tau = \frac{t}{t_f}$, x_0 is the initial position at t = 0, and x_f is the final position at t = t_f. Figure 4-17 provides the 5th order polynomial trajectory to move the robot.

4.5.2.1 *Test description and results*

To evaluate the robot motion, the 5th order trajectory presented is based on the human head trajectory analysis in Chapter 3 was implemented, to evaluate the real-time robot performance. A single axis (the *z* axis) was used to perform a simple, one-dimensional target acquisition task, in order to enable preliminary analysis of the following to be undertaken:

- The evaluation of ALTER performance against varying velocity
- The strategies adopted to successfully achieve a smooth robot trajectory

The robot is commanded to move from a predefined starting position to a target position at a distance 160mm from the starting position with varying test completion time, *T* ranging from 2s to 20s (in increments of 2s). The position and velocity tracking curves for 160mm polynomial trajectory with maximum velocity approximately 150mm/s is depicted in Figure 4-17. The desired position is generated on the QNX PC based on a polynomial function defined in equation (4-3) and is send continuously to the robot controller (desired robot position). Figure 4-17 also proves the effectiveness of the designed synchronization mechanism as discussed in section 4.4.4 as the received desired position command (----- line) is approximately analogous to the generated desired polynomial trajectory. The robot is shown to track very closely the desired position, except during the initial 0.8 second trajectory period when the robot velocity was slowly increased and lags behind the demanded velocity. However, the robot was then moved at velocity slightly higher than desired velocity to reach desired position within allocated time.

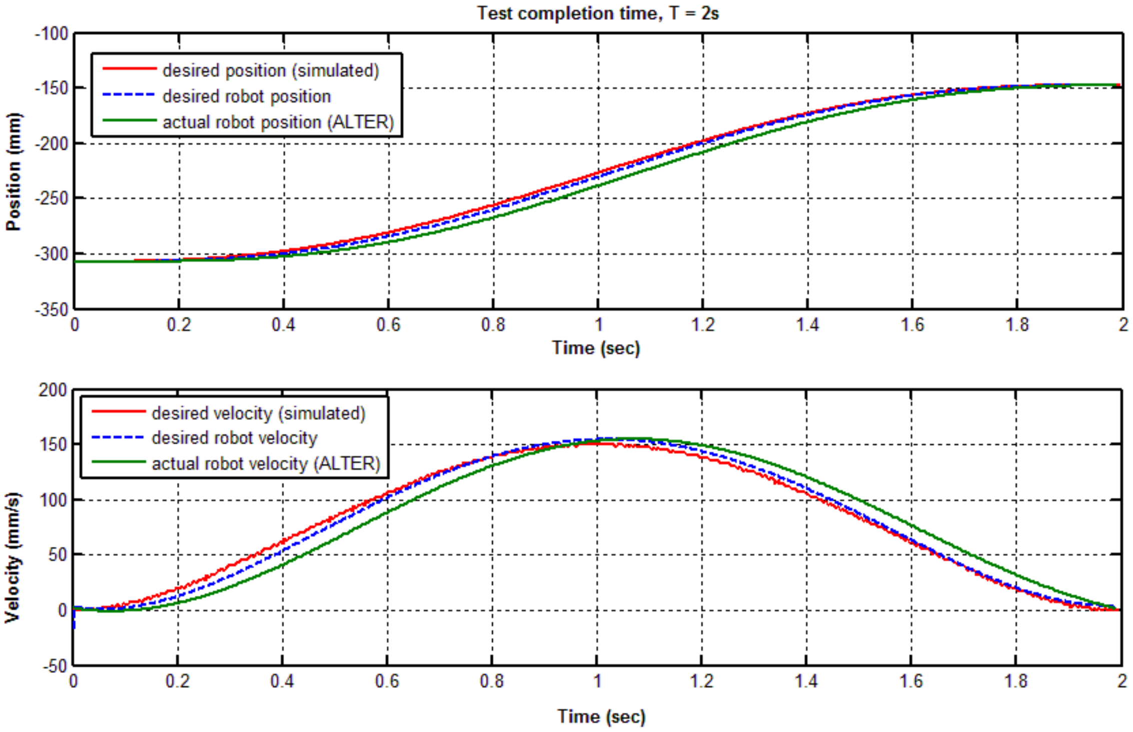


Figure 4-17 Position and velocity profile at *T* = 2 seconds

The results of Figure 4-18 allow easy comparison with a lower robot velocity compared to the Figure 4-17. Figure 4-18 shows that the improved performance with decreasing of the robot velocity. The actual robot position and velocity are approximately similar to the demanded position and velocity profiles which are 160 mm and 20 mm/s respectively. In a comparable manner to Figure 4-17, a small position error is observed during the test with the maximum error is noted in between 0 to 4 seconds due to relatively small calculated incremental positions which is lower than the robot resolution. Nevertheless, the robot manages to reach the demanded position completely within assigned test completion time.

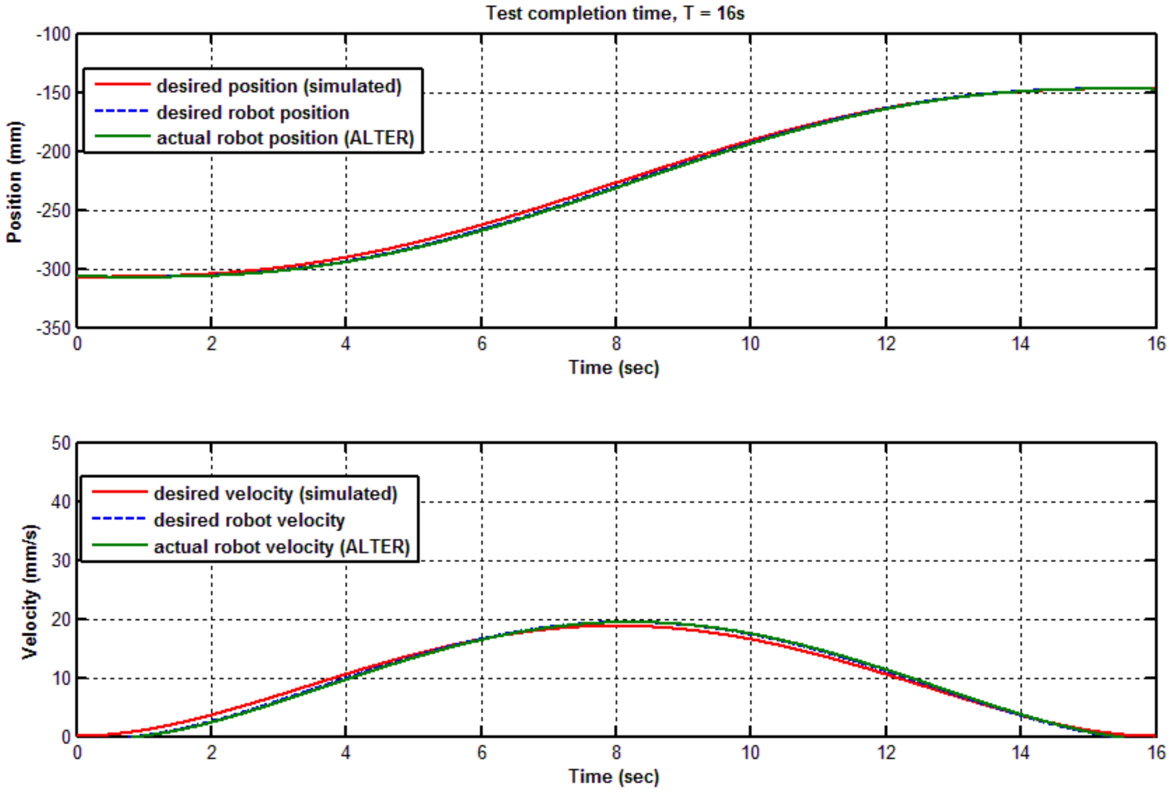


Figure 4-18 Position and velocity profile at *T* = 16 seconds

The results of these position control evaluation tests are too numerous to be included and can be found in Appendix E-4. Figure 4-19 shows a plot of delay time versus fixed demanded robot position for varying maximum velocities. Noted that the maximum delay time for all maximum velocities is ≤ 0.06s, which there has been a gradual decrease in the number of delay time in position between 120mm and 160mm. Whilst, the number of delay time is likely to decline steadily in position between 40mm and 120mm except for the lowest velocity of 20mm/s.

As expected, the maximum velocities decrease with decreasing test completion time, and almost directly proportional at low velocities. To give an idea of the relative performances, the maximum position tracking errors are illustrates in Figure 4-20. The graph approximates the proportional relationship between the robot maximum velocity and position error. Here, it was found the ALTER performance is deteriorated at maximum velocities greater than 50mm/s as the recorded position error indicate large variations (> 3mm).

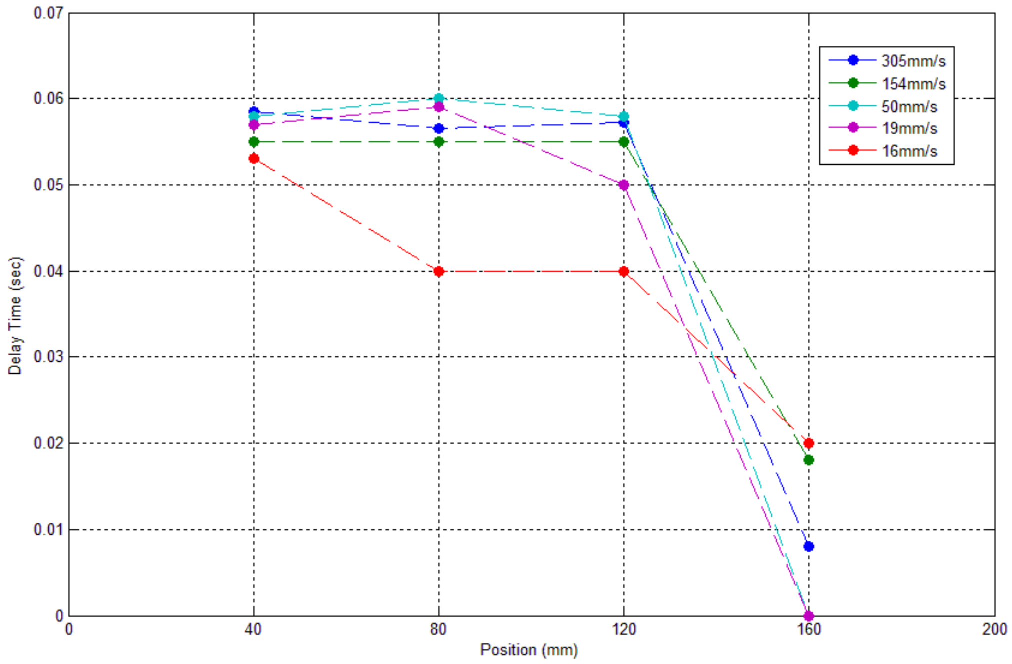


Figure 4-19 Robot position vs. delay time

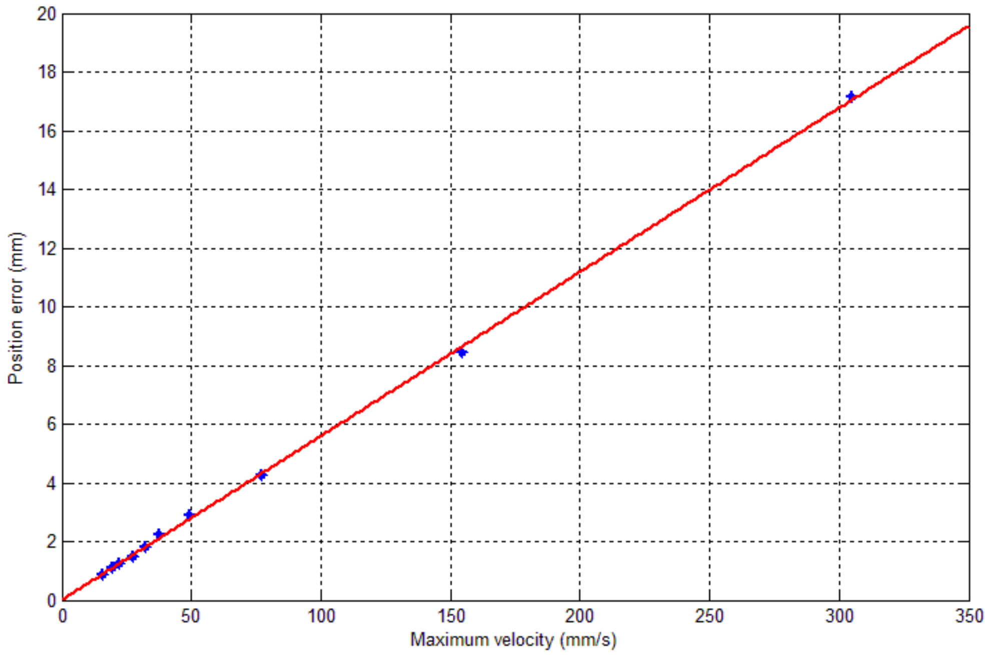


Figure 4-20 ALTER position error tracking vs. maximum velocity

4.6 Guideline of Practical Implementation and Design of Safety System

A major characteristic of TMS robotic system is to strongly interact with human environment. This is very challenging issue in the development of TMS robotic system design as it is understandable that any failure may become very critical. Duchemin et. al [2004] suggest that for a safe robot-human interaction, the robot system must function safely with high reliability. Recent developments in the field of medical robots have led to a renewed interest in safety issues. Numerous studies analyze the constraints that robots have to satisfy and provide general design and manufacturing guidelines and requirements for their medical robot applications [Pierrot et al., 1999; Davies, 2000; Burger et al., 2001; Dombre et al., 2003; Korb et al., 2003; Duchemin et al., 2004]. However, to date there is no evidence of any research that has focused on safety in the TMS robotic system application.

In this section, a safety analysis was conducted in order to identify the constraints that system must satisfy. Then, a specific practical implementation and design guidelines for TMS robot safety system are described in two different levels according to hardware and software.

4.6.1 Safety Requirements

Safety is an important issue in industrial and medical robots. Generally, an industrial application safety is achieved by keeping operators outside the robot's workspace or by shutting down the system if the person comes too close. In contrast, in medical robotic applications it is necessary for medical staff and the patients to be within the robot's workspace. In the TMS application the robot arm is used to hold a coil that is in contact with the subject's head. Hence, the design of a robotic system for use in TMS application poses a challenging safety issue, since there is potential for physical injury if the robot functions incorrectly. Based on medical devices classification guidance the TMS robotic system can be classified as IIb active device which required a higher level assessment [CE, 1998-2009].

As discussed before, the proposed robotic system determines the requirements and functional specifications for each component, thus the definition of characteristics with respect to the safety of the system was conducted. A fault three analysis (FTA) and failure and effects critical analysis (FMEA) were conducted to carefully determine the hazards severity and occurrence probability following by risk minimization step which includes design changes, protection mechanisms and warnings indication as shown in Figure 4-21. The process is continued until the hazards are identified as **ALARP** (as low as reasonably practicable); the hazards allocated represent a combination of the minimal practicable level risks which are considered due to the advantages of accepting the risk as well as the cost of reduction. Finally, a robotic system can be used for clinical testing if the benefit is higher than the residual risk.

4.6.2 Safety Analysis Methods

Proper safety design begins with a risk or hazard analysis. All possible dangers resulting from the robot-human interaction in the system caused by insecure conditions and actions. In recent years researchers have suggested that **Fault Tree Analysis (FTA)** and **Failure Mode and Effect Analysis (FMEA/FMECA)** appear to be the most appropriate techniques for robot system analysis [Dhillon and Fashandi, 1997; Davies, 2000; Burger et al., 2001; Korb et al., 2003; Kazanzides et al., 2008]. **FTA** is a “top-down” approach which identifies potential hazards as origin events. In a deductive manner, the possible causes or fault modes of the next lower functional systems are identified and result in one or more components faults. On other hand, **FMEA/FMECA** is a “bottom-up” analysis in which all possible failures are systematically identified as root events. For each identified failure, the potential hazard, the effect, the detectability and the corrective action have to be listed.

The main difference between the two analysis techniques is that FTA includes the analysis of severity and occurrence probability, but not of detection probability which is estimated by FMEA. Hence, FTA and FMEA can be performed for TMS robotic system because these two methods can complement each other to achieve an optimum result. For example, a branch of the FTA can identify the control system as potential cause for hazards; however, the complete control system including hardware, software and drives was examined in FMEA. The hazard potential can be accurately quantified as shown in Table 4-2.

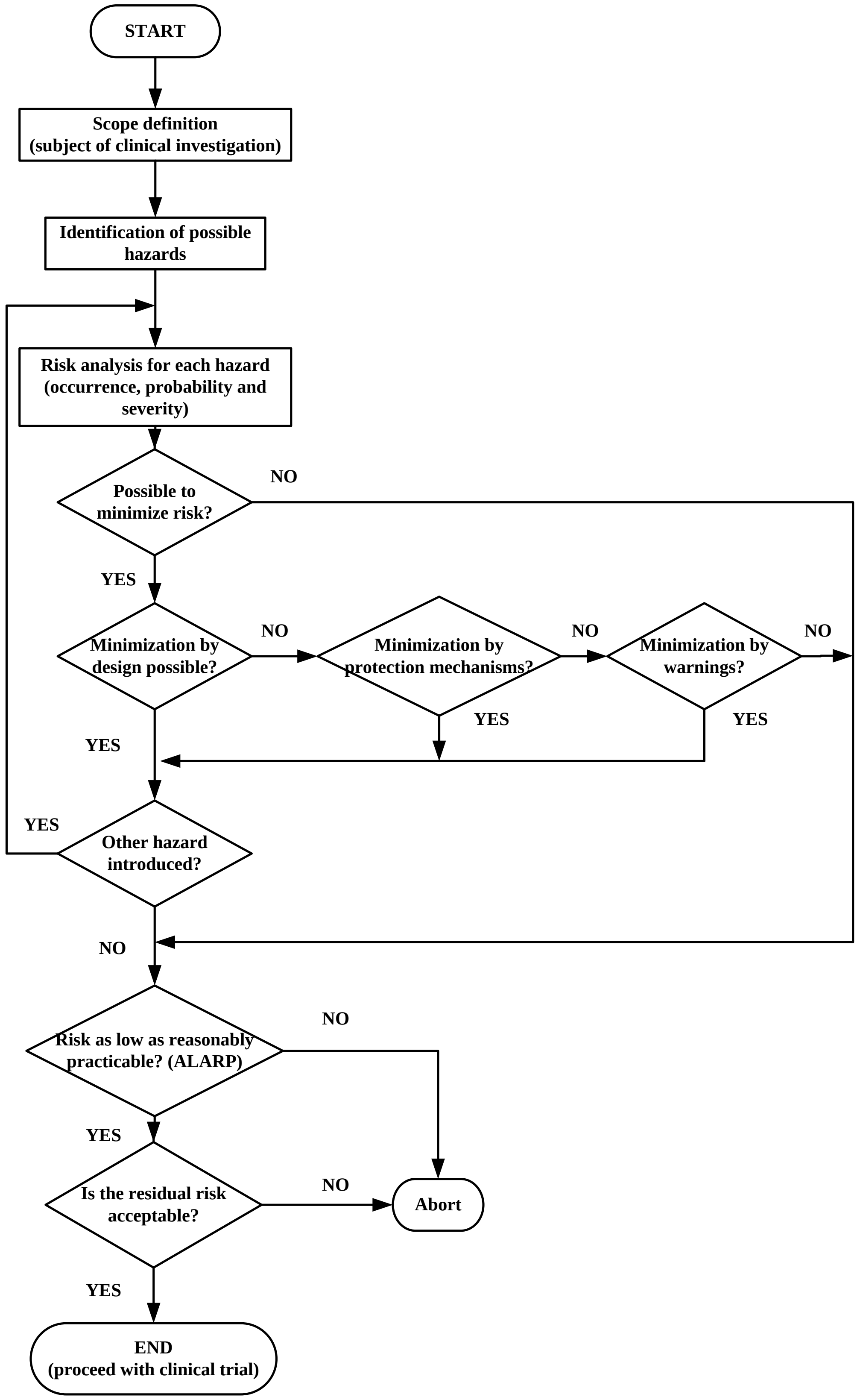

START
Scope definition
(subject of clinical investigation)
Identification of possible hazards
Risk analysis for each hazard (occurrence, probability and severity)
Possible to minimize risk?
NO
YES
Minimization by design possible?
NO
Minimization by protection mechanisms?
NO
Minimization by warnings?
NO
YES
YES
YES
YES
Other hazard introduced?
NO
Risk as low as reasonably practicable? (ALARP)
NO
YES
Is the residual risk acceptable?
NO
Abort
YES
END
(proceed with clinical trial)

Figure 4-21 Flowchart of interactive risk management process applied before clinical trials

Table 4-2: Categorization of occurrence probability [Korb et al., 2005]

Categorization		Occurrence Probability
A	Frequent	In almost every use, occurring short interval
B	Feasible	Occurs regularly, especially induced by other failures
C	Occasional	Occurs at irregular or infrequent intervals
D	Conceivable	Occurrence conceivable
E	Unlikely	Occurrence questionable
S	Systematic errors	Undetermined occurrence that cannot be determine (easily) for software or user failure

Figure 4-22 illustrates an example of the FTA technique applied to one potential hazard occurred in TMS robotic system, i.e incorrect positioning of the coil.

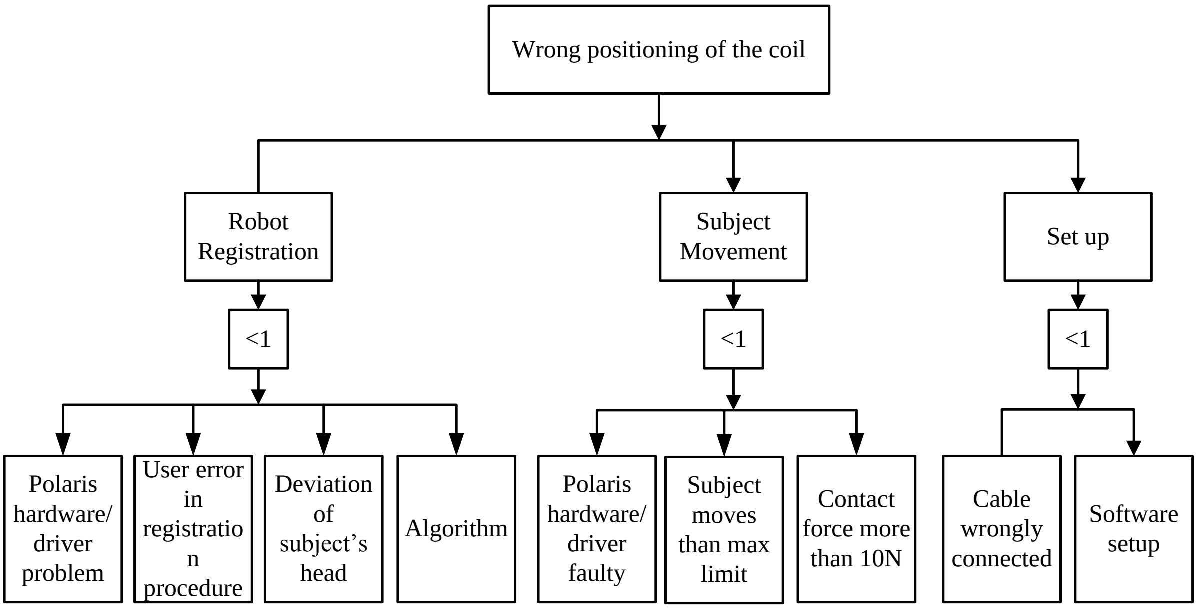

Wrong positioning of the coil
Robot Registration
Subject Movement
Set up
<1
<1
<1
Polaris hardware/ driver problem
User error in registratio n procedure
Deviation of subject's head
Algorithm
Polaris hardware/ driver faulty
Subject moves than max limit
Contact force more than 10N
Cable wrongly connected
Software setup

Figure 4-22 Wrong positioning of the coil

Table 4-3 provides a hazard estimation corresponding to the fault tree of Figure 4-22. The occurrence probability was distinguished according to flowchart in Figure 4-21.

Table 4-3: Hazard estimation corresponding to the fault tree of Figure 4-22. The occurrence probability was distinguished according to Table 4-2.

Code	Sub Hazard	Occurrence Probability
F11	Polaris hardware/driver problem	**S**
F12	User error in registration procedure	**D**
F13	Deviation of subject’s head	**?**
F14	Algorithm	**S**
F21	Polaris hardware/ driver faulty	**D**
F22	Over limit subject movement velocity	**?**
F23	Contact force more than 10N	**D**
F31	Cable wrongly connected	**B**
F32	Software setup	**S**

Note that several sub hazards with question marks are difficult to categorize because insufficient relevant studies are available. It is suggested that an appropriate risk analysis be carried out to increase the reliability of the robotic system, with a special focus on subject movement. The velocity and differences between active (body movements, sneeze, cough) and passive (movement caused by breathing) movements are determined to obtain an appropriate solution for the risk problem. Further research which takes these methods into account should be undertaken in order to establish safety systems for the robotic TMS system.

Furthermore, the safety system for each medical robot is different in which each system is designed particularly for intended purpose [Pierrot et al., 1999; Dombre et al., 2001; Dombre et al., 2003; Poignet et al., 2003; Duchemin et al., 2004]. Following section deals with an outline design of the safety system that needs to be addressed in this work as shown in Figure 4-23. It covers the two main aspects namely hardware and software safety features as presented and shows that such a system can be developed with various existing technologies.

All the robot movement is fully authorized by the neurologist/operator by start button (1) on the robot system with the aid of the graphical user interface. During tele-operation mode movement, a distance limit (2) is programmed in which only allows the operator to manoeuvre the robot arm around a defined TMS procedure zone as shown in Figure 4-24 (a). In the case of a guarded move when the entire robot axes are position control the robot is commanded to move at a low speed of 10mm/s (5) until the force threshold is sensed (4). Involuntary head movements might cause the malfunction to tracking mode and subsequently introduces inevitably harm to the subject. The head movement can be monitored using the Polaris system (3) and any anticipated subject's large movement will stop the system immediately. Finally, by attaching multiple reachable emergency stop buttons (6), this feature enables the operator to stop the robot in case of emergency. The CS8C robot controller software joint limit switch (7) is enabled to limit the robot range of movements. Following sections will discuss the implementation of the safety system in more detail.

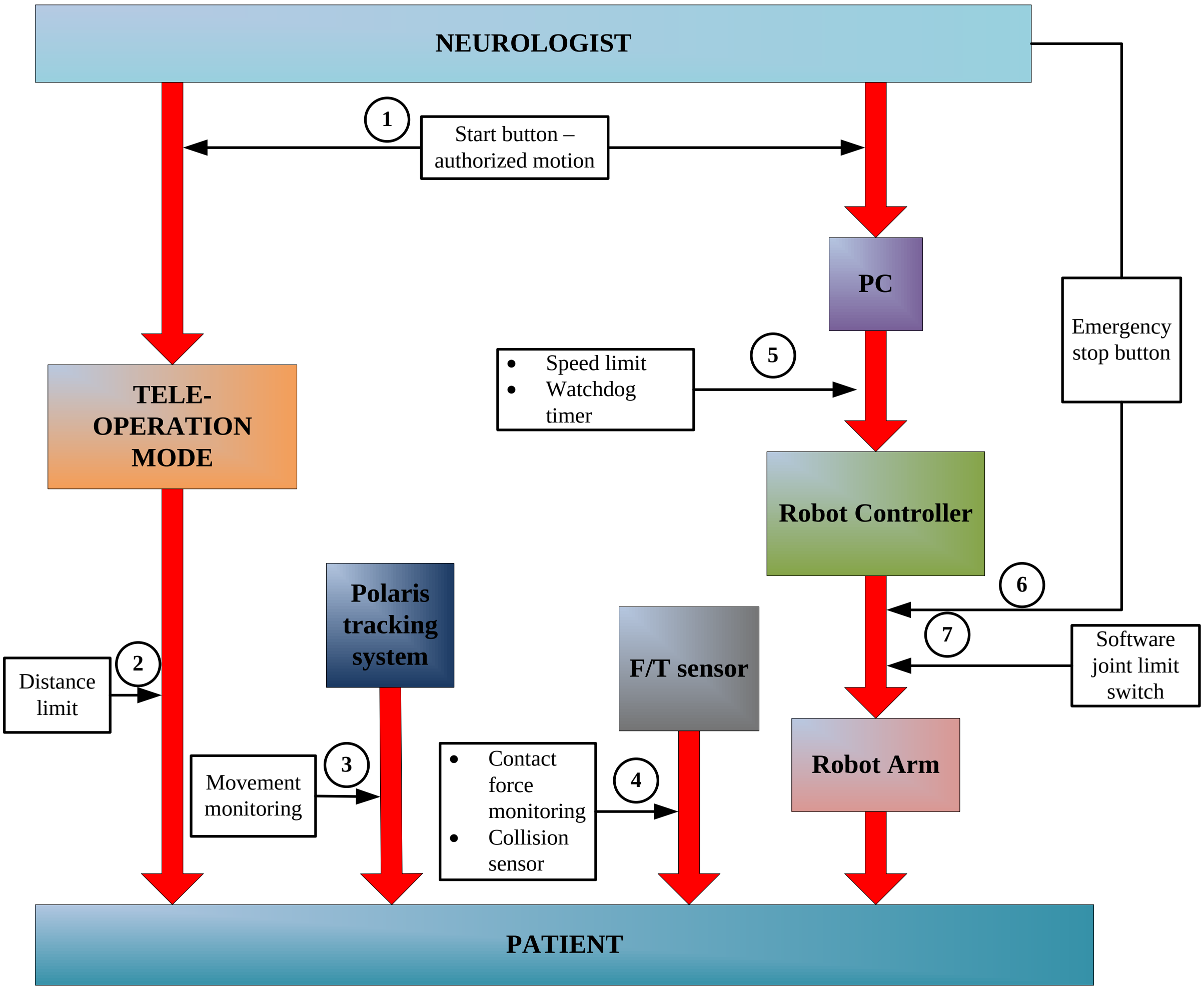


Figure 4-23 Outline of the safety system

4.6.3 Hardware Safety

The practical concepts of intrinsic safety at hardware level involve some aspects of electromechanical design as described below:

- The tele-operation and tracking modes are authorized by pressing a start button on the QNX host PC and robot teach pendant.
- The speed of robot is limited to 100 mm/sec and must be limited to satisfy only the required task to minimise injury if undesired movement occurs.
- It has been suggested that special attention must be paid to provide electromagnetic compatibility (EMC) particularly immunity to electromagnetic fields, electrostatic discharges and noise emission. In addition, all cables must be shielded and integrated to avoid the risk of wrenching or breaking.
- For each external module such as force/torque transducer, host PC and Polaris tracking system is monitored in real-time and emergency stop circuit will be activated if any module malfunction is detected. In addition, multiple emergency stop push buttons with a good accessibility are provided for operator and patient. A mechanical collision avoidance device (Appendix D-1) should also be mounted between the coil and the robot end-effector in order to decouple the coil from the robot if the contact exceeds a safe threshold.
- Figure 4-24 illustrates a three-cable tether system that can be attached to robot joint to avoid (1) waist, (2) shoulder, (3) upper arm, (4) forearm and (5) wrist singularities configuration and satisfy the desired task workspace. The emergency stop switch, S1 is integrated into safety tether system to prevent any possible hazards in the events of power breakdown or emergency stop. The switch S1 is connected to the eye bolt mounted in the middle of upper arm by flexible cable and also connected to the wrist in a fixed position at the robot end-effector. Switches S2 are a side rotary limit switch (GLC series EN50041) (Appendix D-2) and are mounted fixed on the robot joint 1 and is literally spaced, marked as A and B as shown in Figure 4-24 (a). An elongated metal bar is positioned in between switches for the rotation around shoulder axis. The robot normally operates within a spatial zone or volume defined by TMS procedure operation zone. If the robot moves outside the defined operation zone as shown in Figure 4-24 (b) and (c), switches S1 and S2 would be actuated to their open position thus

interrupting robot arm power. The robot arm power can be restored again by the operator manually.

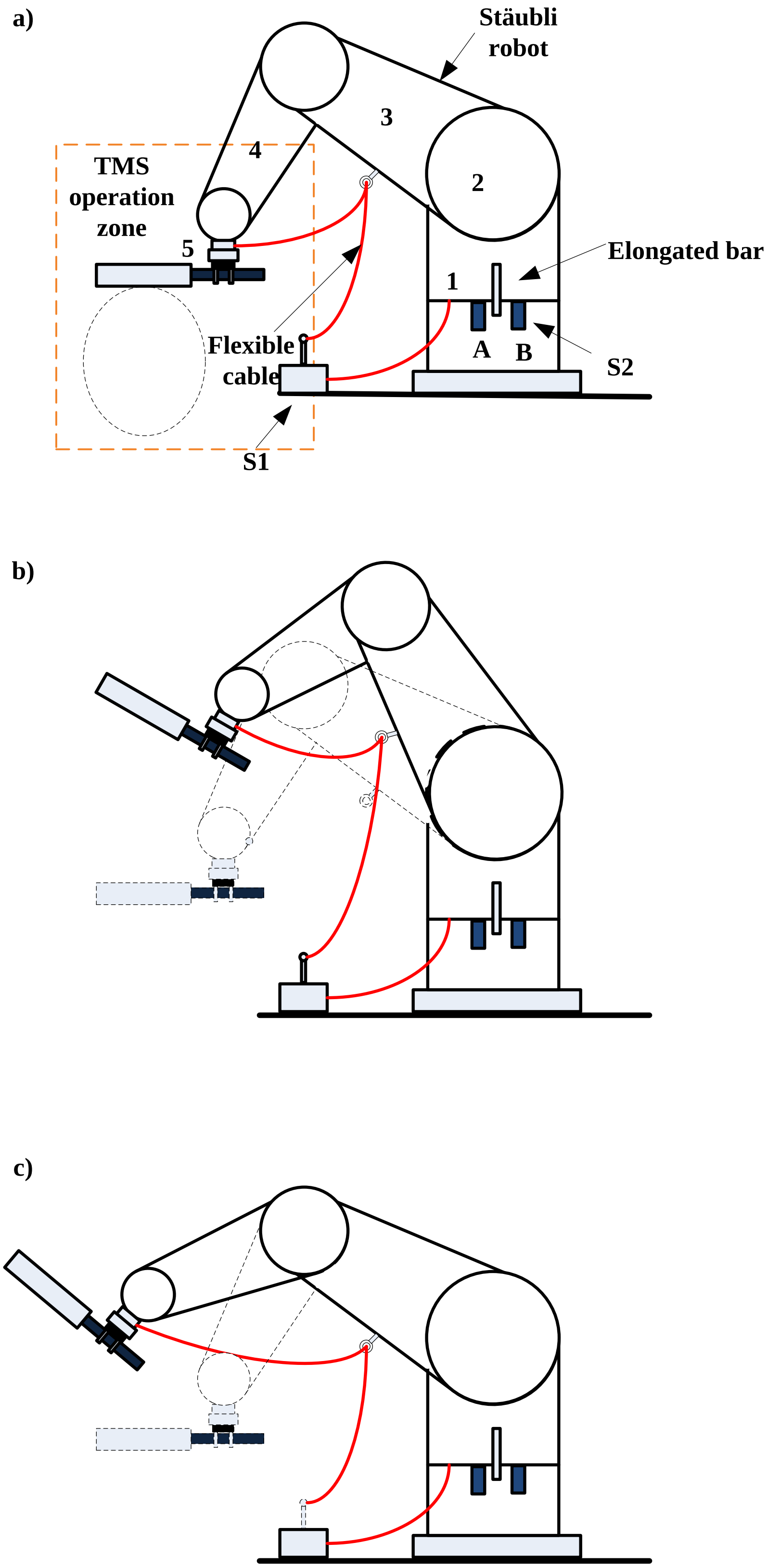


Figure 4-24 A three cable tether system

4.6.4 Software Safety

High-level software has been constructed and the control software is designed to be intrinsically safe. Head and coil velocities, contact force and rate of change of force will be continuously monitored. If any of these parameters drift outside the limit set, the system will be safely shut down. As discussed in section 4.6.2, a FMECA safety analysis was conducted and is observed with every subsystem of TMS robotic system. All possible hazards are classified into three levels of danger: *level 1*–patient or operator death; *level 2*–patient or operator injury; *level 3*–patient or operator problem.

A failure case analysis called multi-creation approach is followed to guarantee safety instead of adding local redundancy systematically. For example, in our case a force sensor is used to monitor the contact between the coil and the subject, subsequently the information is used in the control computer software. A threshold level is also built into the force sensor controller thus; the maximum force that the robot can apply is limited in order to avoid excessive contact force to the subject. Table 4-4 provides example of the multi-criterion approach.

Table 4-4 Multi-criterion approach

Failure	Failure Description and Effect	Effect evaluation on system	Level of hazard
Force > Threshold	Force threshold + 30% > force value > Force threshold ⇩ Force control system failed	Arm stopped	Level 2
Force > 1.3* Threshold	Force value > Force threshold + 30% ⇩ Force control system failed	The collision sensor is activated and arm power off	Level 2
Hardware/ driver faulty	The system becomes in active	Arm deceleration if a motion was going on and current action cancelled	Level 2

Furthermore, additional software joint limits are implemented to reduce the range of robot movement and improve reliability of the robotic system. This feature will extend the lifetime of mechanical components and guarantee a limit even if the mechanical joint limit is broken. The detection of any control error or mechanical breakdown can be done by tracking errors at both Cartesian and joint levels. Generally in real-time programming, variable time consumer procedures should be avoided in order to decrease the execution time of the processes and subsequently increase the sampling frequency in order to react faster in case of emergency.

A software watchdog timer is integrated to the system in order to manage safety from a software point of view. If an intermittent software fault develops in the high-level controller, a signal will be sent to the watchdog to abort the failed process, and subsequently stop and switch off the robot power if the failure is very critical. On the other hand, if the low risk failure occurred, the watchdog timer will initialize the hardware to a ‘safe’ state, and then restart the process in a coordinated manner. While performing some variations of the recovery strategies, the information about the software failure can be collected. The software watchdog can generate a chronologically archived sequence of the failure files. The files could then be used for postmortem diagnostics, and subsequently allows the developer to detect and correct the software problem.

4.7 Summary

This chapter introduces the design of force-controlled TMS robotic system. The system hardware configuration and software architectures that comprise the force/torque data acquisition system, the integration of the subsystem and software programming including the robot communication frame were discussed. One important key performance criteria in this real-time system involves task synchronization between the QNX host PC and the CS8C robot controller which must be maintained within 4ms via TCP/IP socket connection. The ALTER feature available by Stäubli has been implemented in order to control robot movement in real-time, subsequently to allow a stable and robust force control.

In addition, this chapter has set out practical guidelines to establish safety requirements in TMS robotic system applications as well as to identify standard requirements and regulations. The following conclusions can be drawn from the present studies:

- The MDD has to be followed in order to receive CE marking – this is mandatory for commercial medical devices in the European Union.
- It is considered that TMS robotic system can be classified as a medium risk active device.
- It can be concluded that risk analysis is an important part of the safety concept of the TMS robotic system and a key requirement of the MDD.
- A FTA and FMEA/FMECA analysis have been conducted during the development of the TMS robotic system to identify any possible hazard which leads to implementation of the safety system.

This research has thrown up many questions in need of further investigation and work to develop safety systems in the TMS robotic system. In reviewing literature, no data was found on the safety of TMS robotic system. However, a number of possibility safety methods as discussed in other existing medical robot have been applied to the TMS robotic system application. Following chapters will discuss and analyze 1 DOF force-controlled robot system based on this developed TMS robotic system. Subsequently, a series of analysis and simulation are carried out to study the system stability based on this chapter analysis.

CHAPTER 5

SINGLE DEGREE OF FREEDOM ANALYSIS, MODELLING AND SIMULATION

5.1 Introduction

In a TMS robotic system, force control is necessary to maintain the coil in contact with the subject’s head in a more precise manner. If the robot is a purely position-based system, an accurate model of the mechanism and knowledge of the exact location of the head is required, which is impossible to achieve without constraining it in some way. Thus, a need for force feedback control to monitor and control the contact force can provide an effective improvement in system precision and accuracy.

In the TMS robotic system, there is only one coordinate that is constrained using force control, which is the axis normal to the contact environment. This is referred as a one degree of freedom force control which is assumed decoupled from other axes. The implementation of the force-controlled TMS robotic system can be considered as a position-based force controller in which the applied force is compensated at the end-effector by adjusting the position of the robot in real-time.

Figure 5-1 illustrates the schematic diagram of the force-controlled system. Note that the environment position may be changing and the controller is required to maintain the desired contact force while attempting to track the movement of a ball, used to simulate the subject’s head. A PUMA 560 robot is programmed to move the ball.

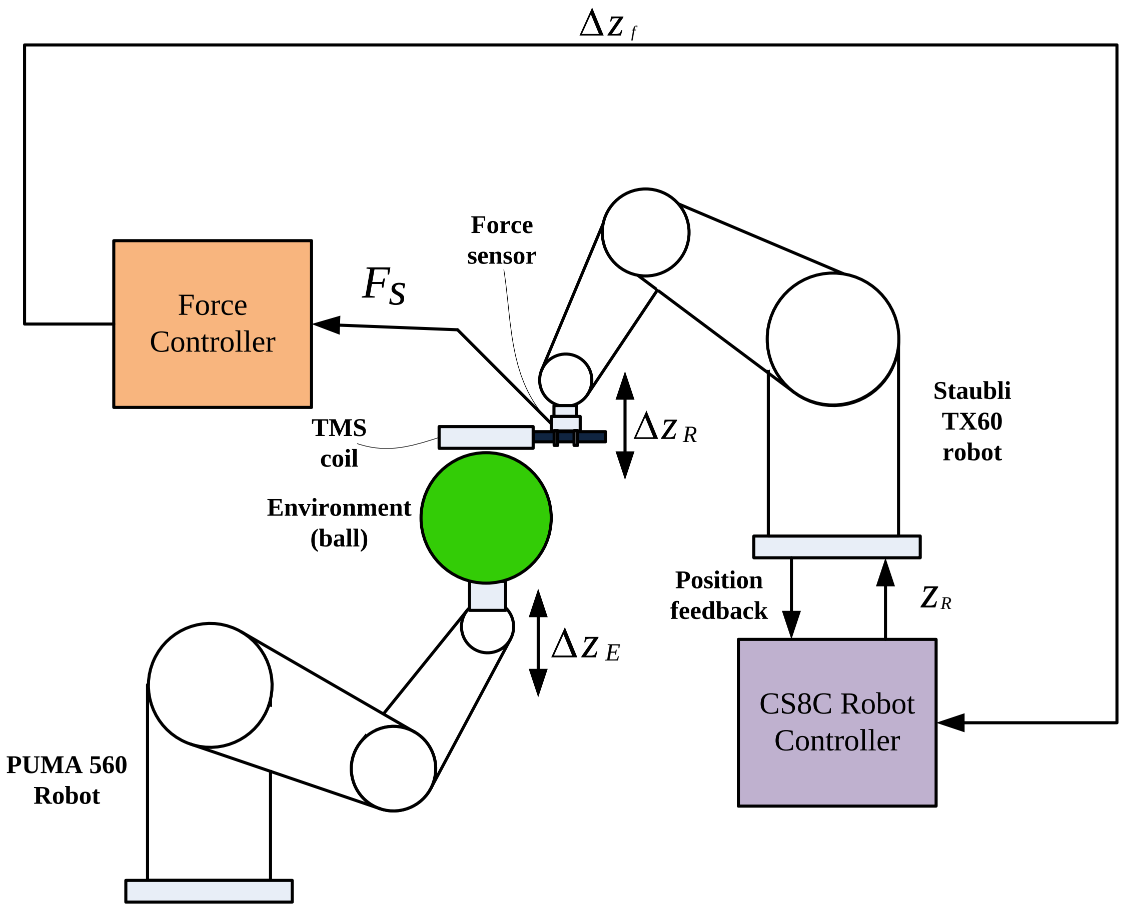


Figure 5-1 The force-controlled TMS robotic system. PUMA robot system is used to simulate motion of the environment (ball).

The TMS robotic system comprises the robot positional servo subsystem and force control subsystem. Assuming that the robot remains in contact with the environment while maintaining a low contact force (of the order of 5N), if the environment is allowed to move without loss of contact, this relative displacement will result in a contact force error to exist between the robot arm and the environment. Robot arm position is adjusted to compensate this error as well as to track the environment movement.

In the case of external hybrid control, the internal position servo loop is maintained and the robot system becomes the ‘open loop’ plant for the force control loop. The major challenge in the development of robot force control is the performance limitation inherent from its implementation. For instance, in order to apply a fast force trajectory, the closed loop bandwidth must be increased by increasing the control gain. However, the limitation always exists in which increasing the gain beyond the maximum value will introduce instability to the system. The trade-off between the stability and performance results in a sluggish closed-loop system. From this point of view, it is important to understand the stability of the system.

This chapter describes the design of a single degree of freedom model to study the factors affecting stability of a force-controlled TMS robotic system. This analysis concentrates on the characteristics and behaviour of the external force control approach in terms of its time and frequency response. The effects of force feedback gains, environment stiffness and the influence of environment position changing in the control loop are presented.

5.2 Modelling

The single axis system can be modelled as a linearized mathematical model of lump masses with the assumption that any resulting movement tends to be limited and consequently linear. The robot arm is modelled as a rigid body mass M_r and M_c mass represents TMS coil, as shown in Figure 5-2.

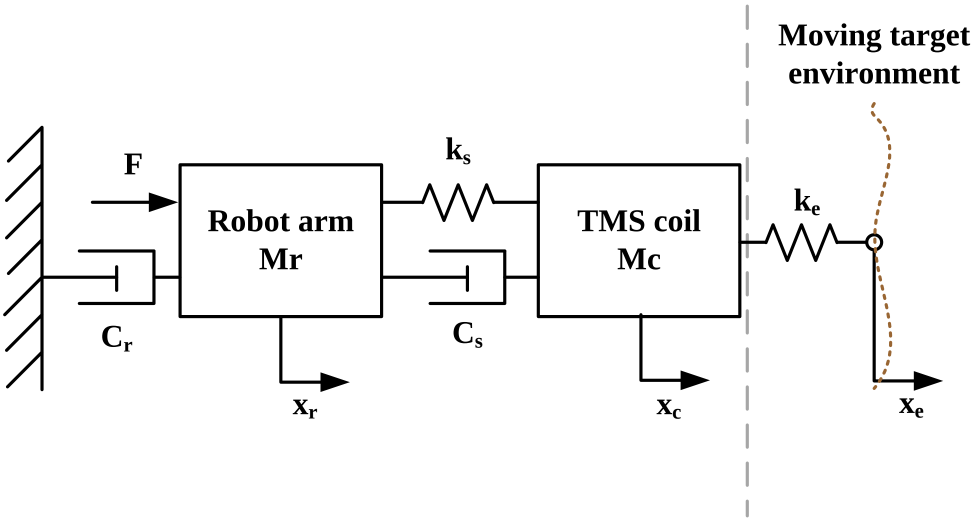


Figure 5-2 Basic mechanical model of the single force controlled degree of freedom

The sensor and environment stiffnesses are represented as k_s and k_e, respectively. Viscous damping is represented by C_r and C_s. The environment is shown as a moving and compliant with position variable of x_3. Mass displacements with respect to M_r and M_c are given as x_r and x_c. The sensed force is determined by measuring the relative displacement of (x_r – x_s) multiplied by the sensor stiffness, k_s. *F* represents the controlled actuator force, delivered by the actuators. The free body diagram of the model is given in Figure 5-3.

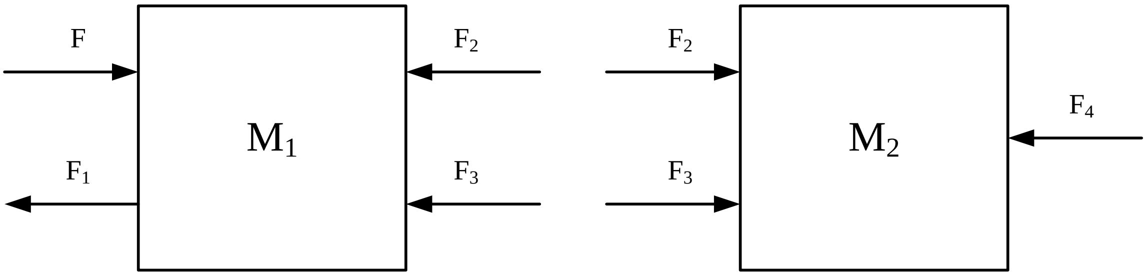


Figure 5-3 Free body diagram, where $F_1 = C_\mathrm{r}\dot{x}_\mathrm{r}$, $F_2 = k_\mathrm{s}(x_\mathrm{r} - x_\mathrm{r})$, $F_3 = C_\mathrm{s}(\dot{x}_\mathrm{r} - \dot{x}_\mathrm{c})$ and $F_4 = k_\mathrm{e}(x_\mathrm{c} - x_\mathrm{e})$.

The equation of motion of the robot, mass M_1 as shown in Figure 5-2 can be written as:

$$M_r\ddot{x_r} = F - C_r\dot{x_r} - k_s(x_r - x_c) - C_2(\dot{x_r} - \dot{x_c}) \tag{5-1}$$

and for coil mass, M_c as:

$$M_c\ddot{x_c} = k_s(x_r - x_c) + C_s(\dot{x_r} - \dot{x_c}) - k_e(x_c - x_e) \tag{5-2}$$

Taking the Laplace transform, equation (5-1) and (5-2) becomes

$$(M_r s^2 + (C_r + C_s)s + k_s)x_r = F + (C_s s + k_s)x_c \tag{5-3}$$

$$\big(M_c s^2 + C_s s + (k_s + k_e)\big)x_c = (C_s s + k_s)x_r + k_e x_e \tag{5-4}$$

The force controller is designed to apply a desired contact force on a moving target environment. In TMS system an external force control approach was chosen to drive the robot arm to maintain the desired contact force by adjusting the robot position, x_f in which is calculated as follows;

$$x_f = k_p(F_d - F_s) \tag{5-5}$$

where k_p is force controller proportional gain.

Combining (5-3), (5-4) and (5-5) gives the closed loop force control relationship

$$F_s(s) = T(s)F_d(s) + D(s)X_e(s) \tag{5-6}$$

Equation (5-6) shows that the environment position/motion, x_e acts as a disturbance that perturbs the desired force, F_d. T(s) and D(s) are force tracking and environment position transfer functions respectively.

where

$$T(s) = \frac{F_s}{F_d} = \frac{k_p k_e G(s)}{1 + k_p k_e G(s)} \tag{5-7}$$

$$D(s) = \frac{F_s}{x_3} = \frac{k_e}{1 + k_p k_e G(s)} \tag{5-8}$$

and

$$G(s) = \frac{C_s s + k_s}{\begin{gathered} M_r M_c s^4 + (M_c(C_r + C_s) + M_r C_s)s^3 \\ +(M_c k_s + M_r(k_s + k_e) + C_r C_s)s^2 \\ +(C_r(k_s + k_e) + C_s k_e)s + k_s k_e \end{gathered}} \tag{5-9}$$

An indication of the system behaviour can be obtained by studying the open loop transfer function with F_d and x_e are equal to zero. It can be derived as

$$G_{O/L}(s) = \frac{(C_s s + k_s)(k_p k_e)}{\begin{gathered} M_r M_c s^4 + (M_c(C_r + C_s) + M_r C_s)s^3 \\ +(M_c k_s + M_r(k_s + k_e) + C_r C_s)s^2 \\ +(C_r(k_s + k_e) + C_s k_e)s + k_s k_e \end{gathered}} \tag{5-10}$$

Figure 5-4 illustrates the equivalent block diagram of the single-axis TMS force controlled system.

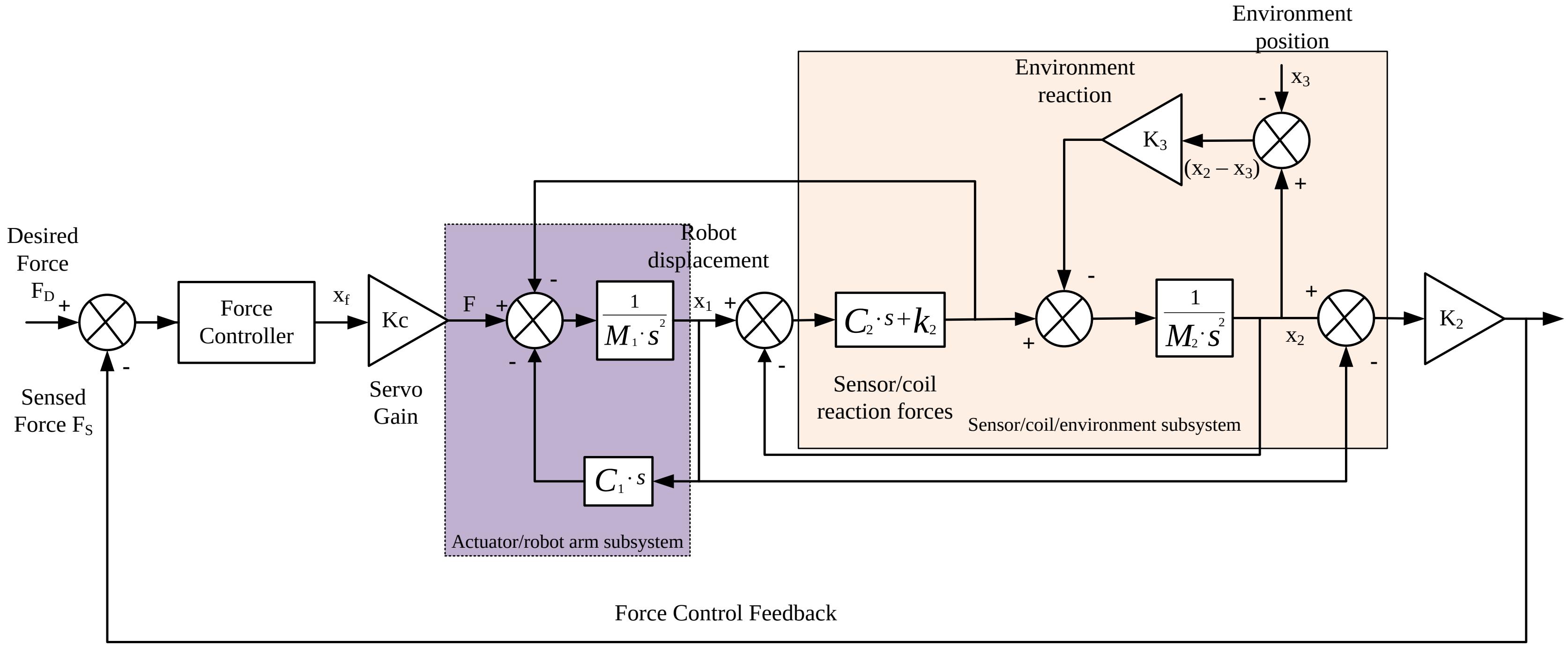


Figure 5-4 Block diagram representing the single-axis TMS force-controlled system

5.3 Single DOF force-controlled TMS robotic system simulation

The experimental robot as described in Chapter 4 was used as the basis for choosing the physical values of the simulation parameters. The model parameters of Table 5-1 and the simulation block diagram (Figure 5-5) are only approximations to those of the TMS robotic system in Figure 5-1. The MATLAB® System Identification Toolbox™ software was used to estimate the mathematical model parameters to fit measured data from the robot dynamics system and was validated after each estimation process to obtain the best dynamic model for the force-controlled robotic system.

The construction of the model involves three basic entities:

- A data set – the input-output data was recorded during particularly designed identification test experiments. A variant of force/position test response were conducted to identify the actuator/robot arm and sensor/coil/environment subsystems parameters.
- subsystems parameters; A set of candidate model – a model transfer function with unknown physical parameters was constructed from established transfer function in section 5.2.
- Validate and examine model output – the model-output plot was use to check how well the model output matches the measured output. In addition, the model also can be validated by examining the behaviour of its residuals using Residual Analysis plot.

Provided the parameters are 'realistic' to a correct order of magnitude and relative values to the real system, the following Table 5-1 provides the estimation of robot system physical parameters.

Table 5-1 Nominal estimation robot parameters

Robot arm mass, M_r	25kg
Coil mass, M_c	2kg

Robot damping, C_r	20N/mms^{-1}
Sensor damping, C_s	12N/mms^{-1}
Sensor stiffness, k_s	10.65N/mm
Servo gain, k_c	7.78N/Nmm
Proportional controller gain, k_p	variable

The MATLAB® Simulink block diagram for the single-axis force controlled system is shown in Figure 5-5.

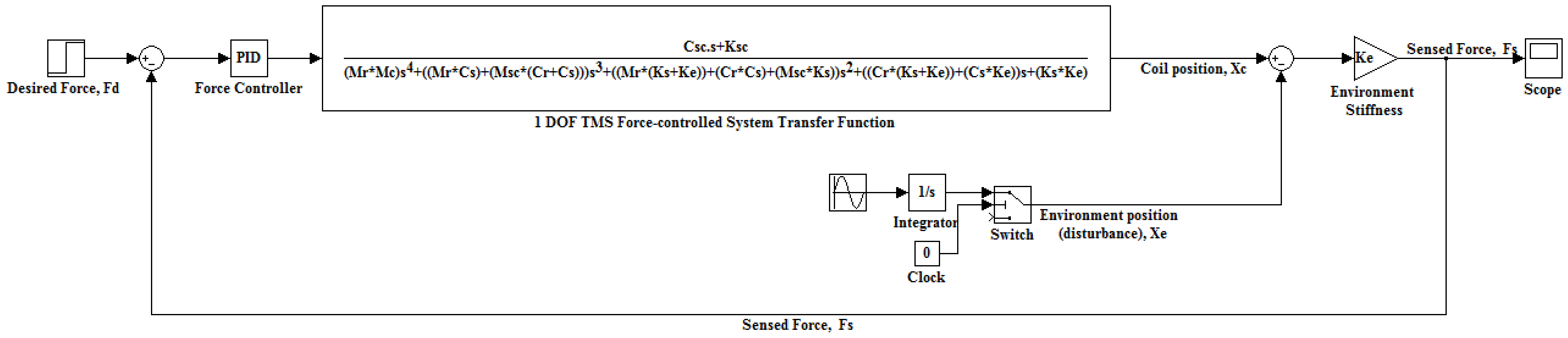

Desired Force, Fd
PID
Force Controller
Csc.s+Ksc
(Mr*Mc)s4+((Mr*Cs)+(Msc*(Cr+Cs)))s3+((Mr*(Ks+Ke))+(Cr*Cs)+(Msc*Ks))s2+((Cr*(Ks+Ke))+(Cs*Ke))s+(Ks*Ke)
1 DOF TMS Force-controlled System Transfer Function
Coil position, Xc
Ke
Environment Stiffness
Sensed Force, Fs
Scope
1/s
Integrator
0
Clock
Switch
Environment position (disturbance), Xe
Sensed Force, Fs

Figure 5-5 Matlab Simulink model of 1 DOF Force-controlled TMS System

5.4 Effect of parameter variation on stability

5.4.1 Stability analysis

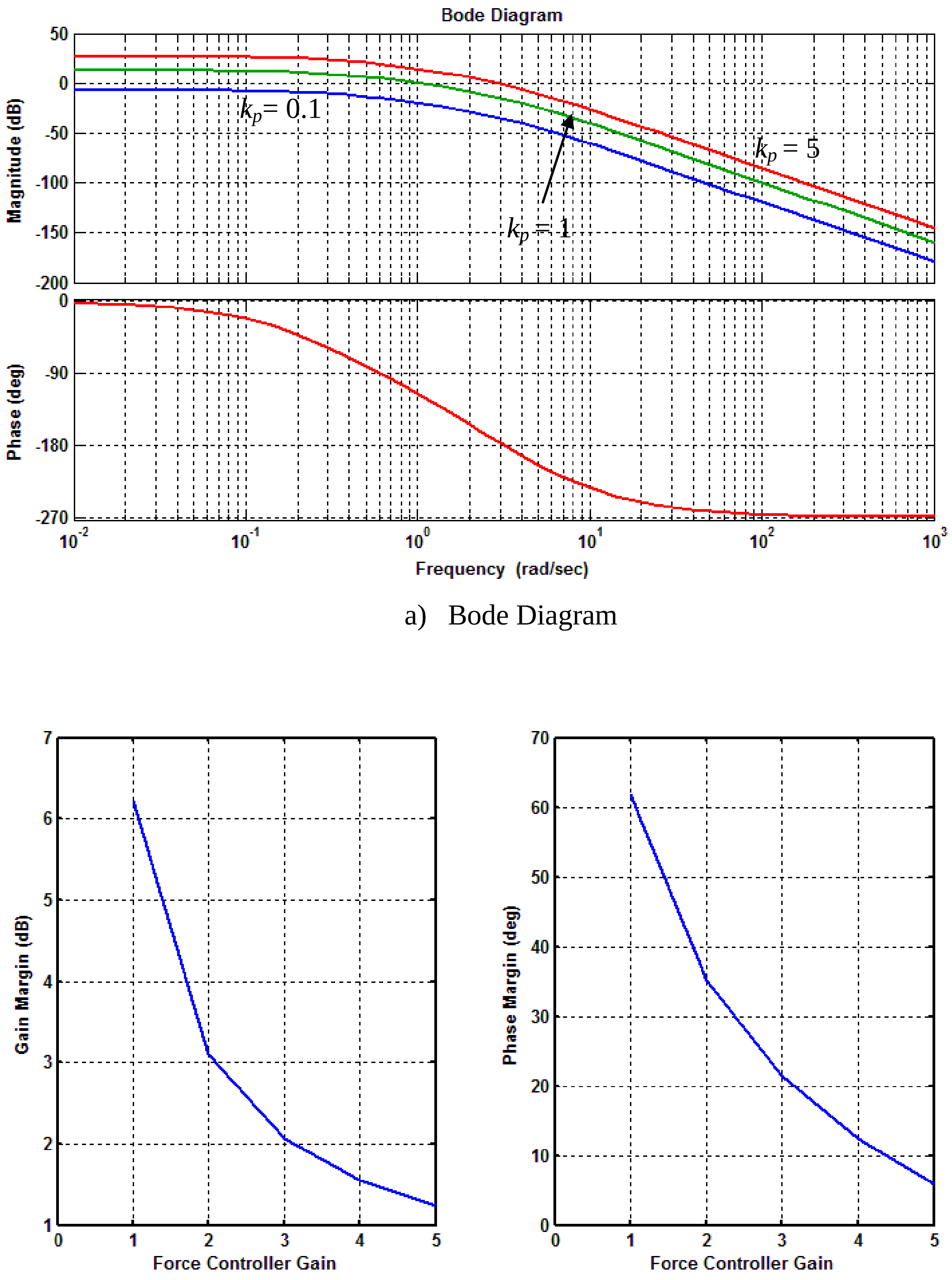
The stability of the above derived closed loop transfer function (5-6) depends on the locations of the poles which must be negative real parts to prove stability. In this study, the Bode diagram is used as a tool for the force control system stability analysis. The diagram indicates that the stability can be measured directly through the gain and phase margins. Phase margin is the amount of additional phase lag at the gain crossover frequency that can bring the system to instability. The phase margin also measures the system tolerance to the time delay in which the system will become unstable if the time delay is greater than 180/ω_{pc} (where ω_{pc} is the frequency where the phase shift is 180°). Gain margin is complementary to the magnitude of open loop gain at the frequency where the phase angle is -180°. The gain and phase margins should be definite positive values against any system parameter variations to ensure stability. The two values bound the behaviour of the closed-loop system near the resonant frequency, and for satisfactory performance, the gain margin should be greater than 6 dB and the phase margin should be between 30^o to 60^o [Bolton, 2002]. This approach provides a convenient means to design the control system as well as adjusting the constant gain for the system. In addition, the transient time response is also discussed in the following section to prove the frequency analysis response and correlation between these two responses.

5.4.2 Varying Force Controller Gain, k_p

The following sequence of simulation results illustrates how a variation in proportional force controller gains k_p, influenced contact force response. The system is based on the simulation model described in Figure 5-2 in which the environment position is assumed stationary, $x_e = 0$. Using the open loop transfer function as given in equation (5-10), the marginal stability of the system is analyzed. Figure 5-6 shows the Bode diagram analysis of an environment stiffness of 10N/mm, which approximates to the ball stiffness in the experimental test when the force controller gain is varied.

The diagram shows the system is stable as the k_p was set to 1. The open loop gain is less than unity or below the 0 dB (20 logs 1) line when the phase is -180°. As can be seen from the graph the gain and phase margins are definite positive, 6.2dB and 61.8° respectively. The gain margin specifies that the system gain can be increased by 6.2dB before instability occurs, as the gain cross-over frequency is less than phase cross-over frequency. Figure 5-7 illustrates the corresponding closed-loop step transient response for F_d = 5N which exhibits about 20% overshoot and minimal oscillations, with a rise time of approximately 2 seconds.

Increasing k_p further to 5 shifted the magnitude curve upward without affecting the phase curve. This caused the gain cross-over frequency to be higher which results in smaller gain and phase margins as shown in Figure 5-6. This implies that the system has become less stable as illustrates in Figure 5-7 by the poorly damped oscillation. On the other hand, the low force controller gain (k_p = 0.1) is associated with the overdamped transient response and high settling time. The phase margin approaches infinity as k_p is decreased beyond 0.1.



a) Bode Diagram

b) Gain margin vs k_p

c) Phase margin vs k_p

Figure 5-6 Bode Diagram – Varying Force Controller Gain, k_p

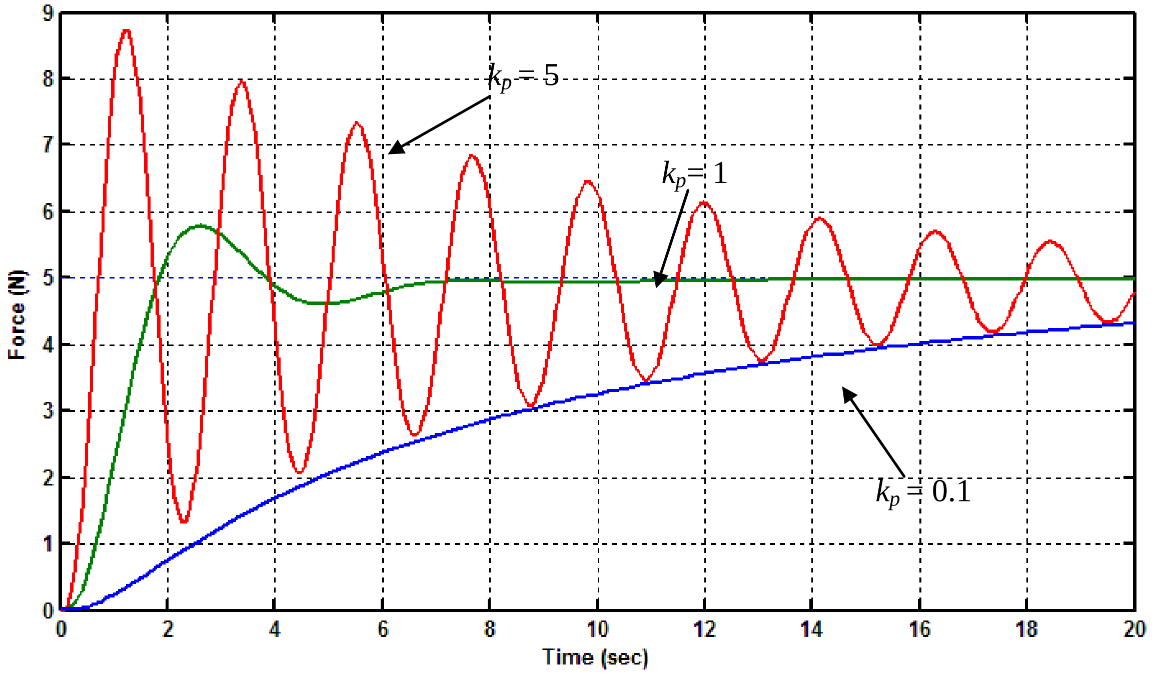


Figure 5-7 Transient time response

5.4.3 Variation in Environment Stiffness, k_e

In the TMS application, the subject's head stiffness may vary from person to person and it is therefore important to study this aspect. The same force controller gain as the previous section, was again used here, where k_p =1. In the external force control approach, if the robot arm is not constrained (k_e = 0), the robot will move at a velocity proportional to the desired contact force. Figure 5-8 shows the minimum phase margin value in unconstrained condition is lower than 90^o compared to the constrained condition. The gain and phase margins of the system are 32.8 dB and 74.95° respectively which implies that the system is stable during free space movement. On the other hand, if the robot system is in contact with the environment (k_e = 10N/mm), this results in a decrease in the gain and phase margins as shown in Figure 5-8.

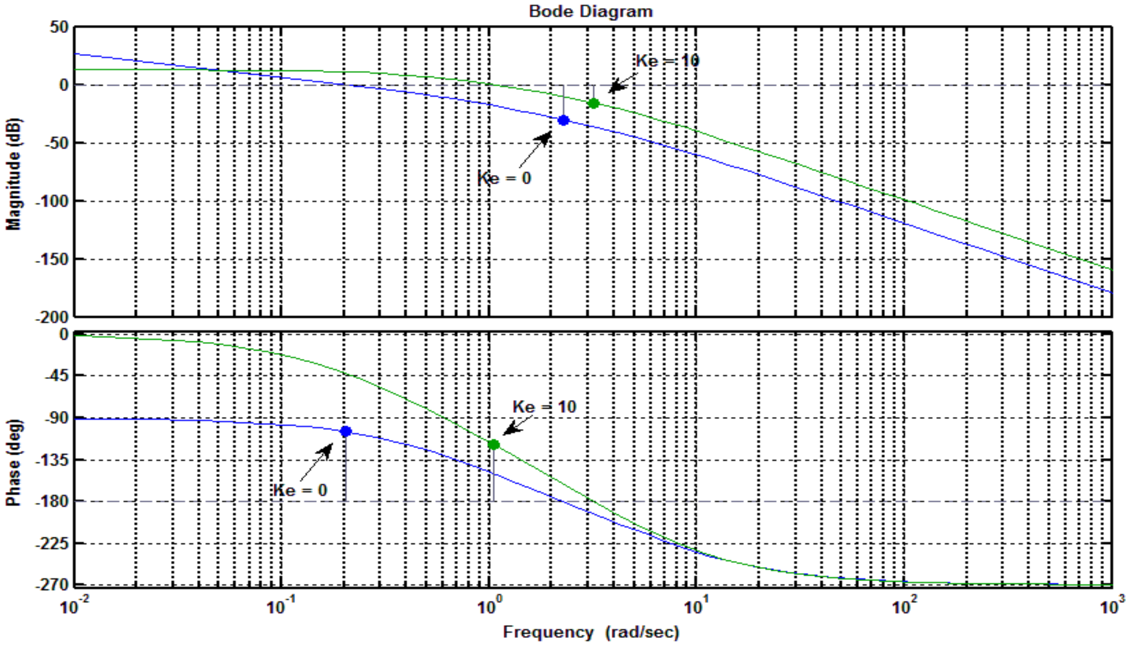


Figure 5-8 Bode Diagram of k_e = 0 N/mm and k_e = 10 N/mm

To see the effect of the environment stiffness to the system, the stiffness was varied between two extreme values which represent a soft environment (1N/mm) and stiff environment (100N/mm) respectively. Conveniently the mid-range is at k_e = 10N/mm, the same experimental test's environment stiffness. All other control parameters including force controller gain and desired force are identical to the test response in Figure 5-8. The system reaction to an increase of k_e is similar to that of increasing force controller gain in which it shifts the magnitude curve without affecting the phase curve as observed in Figure 5-9.

Increasing k_e to 100N/mm shifts the 0 dB axis down by approximately 20 dB resulting phase and gain margins to -12.1^{o} and -4.14 dB respectively. These negative margins indicate that the system is unstable. This is also confirmed by the value of gain cross-over frequency which is higher than the phase cross-over frequency. This result is unacceptable during experimental work which can damage the target environment and the robot system itself. To achieve better stability, the magnitude curve can be shifted downward by reducing the force feedback gain.

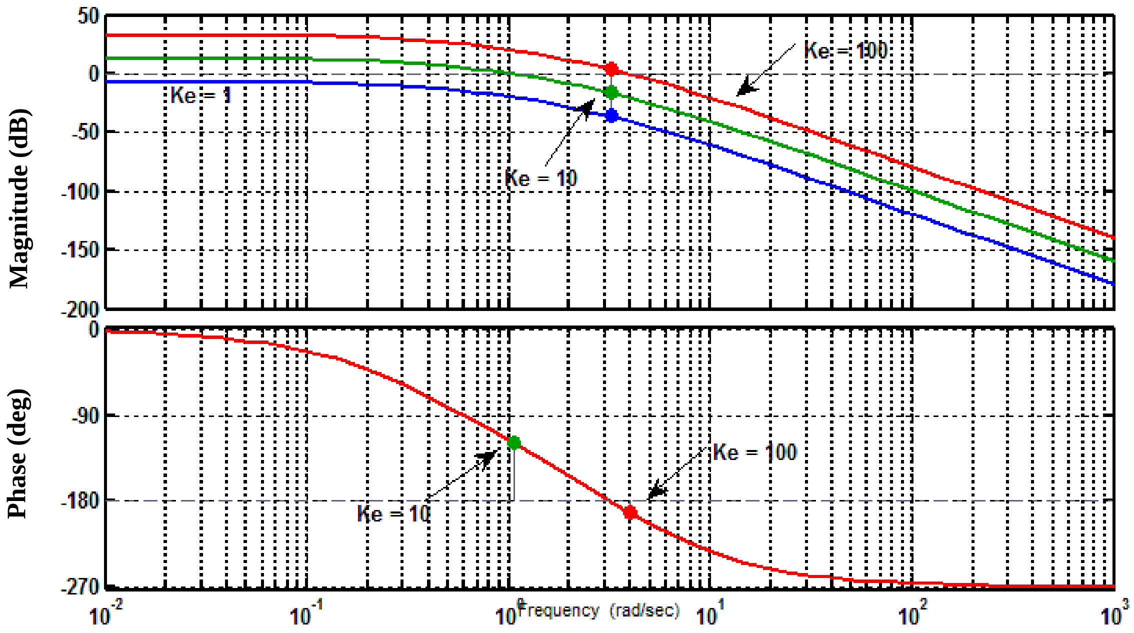


Figure 5-9 Bode Diagram – Varying Environment Stiffness, k_e

It is also observable that a decrease in environment stiffness is accompanied with undesirable response. A softer environment (k_e = 1N/mm) yields large steady-state error as shown in Figure 5-11. To achieve better performance, the magnitude curve can be shifted upward by increasing the force controller gain to 3.2. Figure 5-11 shows a satisfactory response with rise time at 2 seconds and approximately 16% maximum overshoot. The simulation results, while preliminary, suggests that for a softer environment, a higher force controller gain is necessary to obtain a desired performance. On the other hand, the force controller gain has to be reduced for high environment stiffness, k_e.

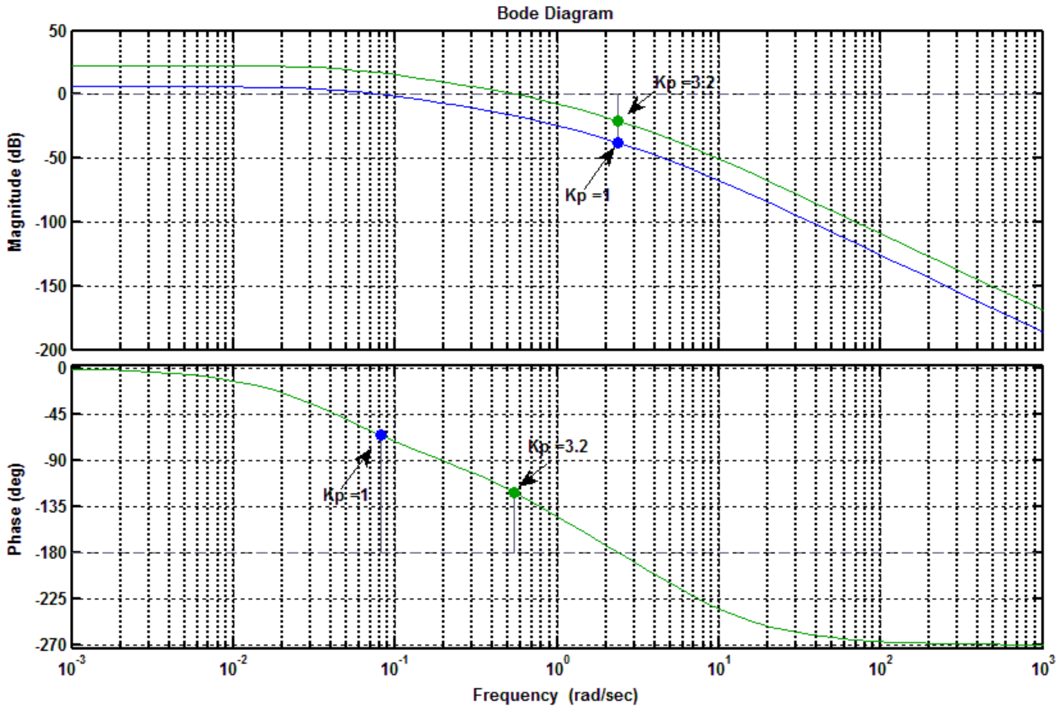


Figure 5-10 Gain and Phase Characteristic of k_e = 1N/mm

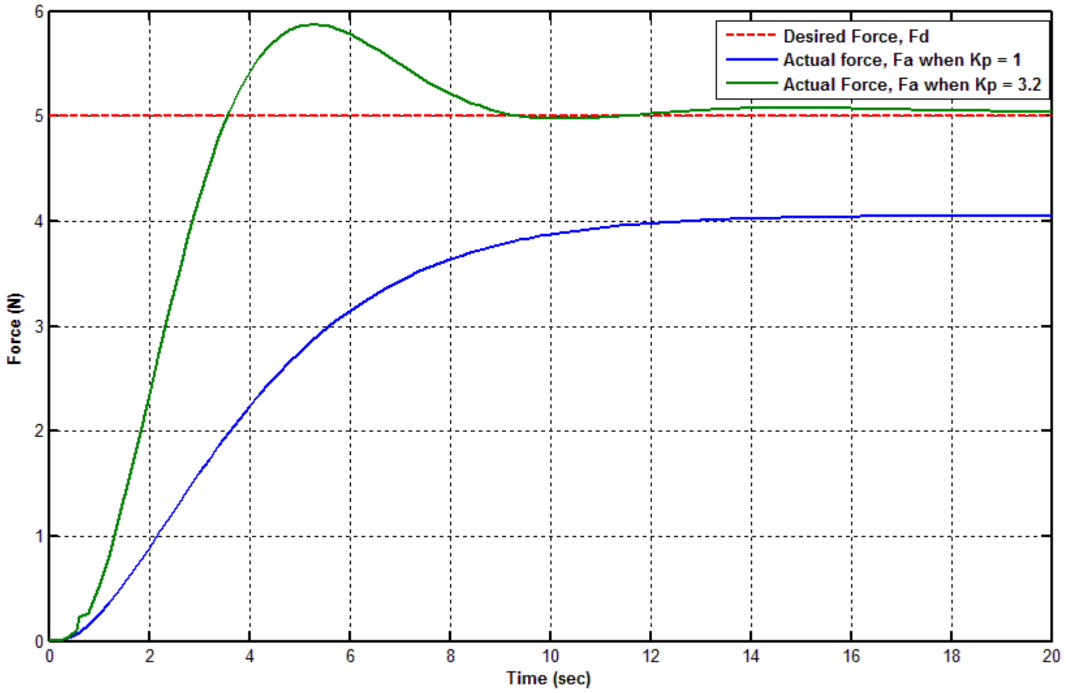


Figure 5-11 Transient response of k_e = 1 N/mm

5.5 Effect of positional disturbance (moving environment)

When a robot interacts with a moving environment, the variations of environment position can be regarded as a positional disturbance to the force control system. So far, the analysis (section 5.3) assumes that the environment is static and the measured contact force is proportional to the robot displacement, x_2. However, if the environment is moving, the resultant contact force imposed on the environment can be assumed as a function of the relative displacement caused by the environment movement.

In this condition, the robot system is commanded to follow a desired position x_d which is a combination of the environment position x_3 and the deformation position that is required to maintain the desired force, x_f as shown in following equation

$$x_d = x_f + x_3 \tag{5-11}$$

The similar model and force controller gain as described in section 5.2 is again studied and used for the simulation. To demonstrate the force control system behaviour, the simulation is repeated with desired contact force of 5N on contact environment of 10N/mm. Figure 5-12 illustrates the contact force response corresponds to a step input disturbance position towards the coil at 12.6 seconds. The contact force is rapidly increased to 10N and takes 2.6 seconds to restore the response to the desired contact force with slightly oscillations. A similar behaviour can be observed when the environment is moving downward from the coil after 46.4seconds. From the position graph in Figure 5-12, it is apparent that this poor behaviour results in the low value of resultant incremental position at a rate of 2mm/s. To achieve better performance, the faster recovery time can be accomplished by increasing the proportional force controller gain. However, there is a maximum value for the gains, as too high of a gain value can cause the system to poorly oscillate and subsequently entering the system limit cycle. One of the issues that emerges from these findings is if the target environment moves with a higher disturbance position/velocity (x_3 > 0.5 for a sampling period of 0.004s) the system is not able to encounter the disturbance, and the environment is prone to suffer a high contact force (upward movement). On the other hand, if the environment moves in downward direction, an intermittent contact between the coil and environment may occur, this introduces instability to the system.

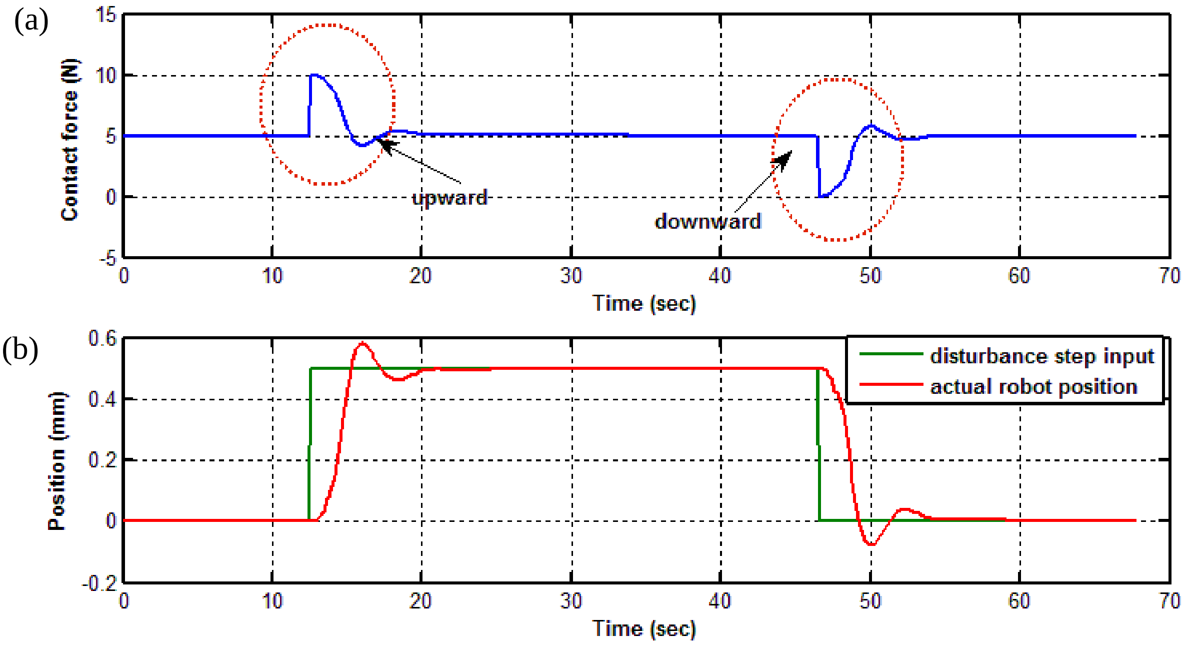


Figure 5-12 a) Contact force response and b) position response for the environment disturbance position step input test

This inadequate behaviour is more problematic in real situation as the target movement is a continuous motion which is approximately about 0.25rad/s to 1.05rad/s in the horizontal, vertical or torsional plane during seated and ‘relax’ position [Ferman et al., 1987; Hagemann et al., 1999; Cavanaugh et al., 2005] as discussed in Chapter 3. Subsequently this section will therefore consider and focus on target environment motion along the normal direction of the surface environment. A sinusoidal function was used to simulate the target motion with variation of frequency that is approximately similar to the above identified human’s head movement.

In order to maintain the desired contact force, it is necessary that the robot system follows arbitrary inputs accurately, thus a faster response is required. To achieve this requirement, the system must have a large bandwidth as it corresponds to a low rise time and fast response. This value can be used to provide a rough estimate of the transient response speed [Ogata, 2009]. Consider the similar simulation model with contact environment of 10N/mm. The system bandwidth frequency for this condition can be found from the Bode diagram in Figure 5-6, which is approximately 0.3rad/s. This value is corresponding to the closed loop magnitude response that equal to -3dB. Satisfactory transient response performance can be obtained if the disturbance position frequency ω_d is lower than system bandwidth frequency, ω_b [Ogata, 2009; Yuen et al., 2010].

Figure 5-13 presents the simulation results obtained from the environment movement at the frequency of 0.25 rad/s (lower than ω_b). To demonstrate the effect of a moving environment, the simulation test in 5.3 was repeated with F_d = 5N. After the contact force reaches the steady-state value of of 5N, the position input is perturbed by 4mm away (downward movement) at t = 3.6 seconds and then moves towards (upward) the coil direction at t = 15.4 seconds. It can be observed that the robot movement is lag behind the environment movement. The robot performance is undesirable, since the lag can cause a large variation of steady-state error for the contact force. The highest contact force error (≈5N) was observed when the environment moves at maximum velocity of 1mm/s. As being discussed before in section 5.4.2, increasing the force feedback gain will reduce the system stability, however unexpected finding was found that increasing the gain from k_p = 1 to k_p = 5 on dynamic target environment will improve the force response as shown in Figure 5-13(a). As can be seen in Figure 5-13(a), the contact force response steady-state error is reduced approximately 60% and 84% for downward and upward movement respectively. The improved force response might be explained by the fact that the robot can significantly tracks the environment motion closely as shown in Figure 5-13(b) with maximum position error of 1.8mm. As an increase in force feedback gain is accompanied with an increase in incremental robot position. Hence it could conceivably be hypothesized that it is possible to maintain desired force response on moving environment by increasing the force feedback control gain. However, caution must be applied, as much higher gain might be introducing instability to the system.

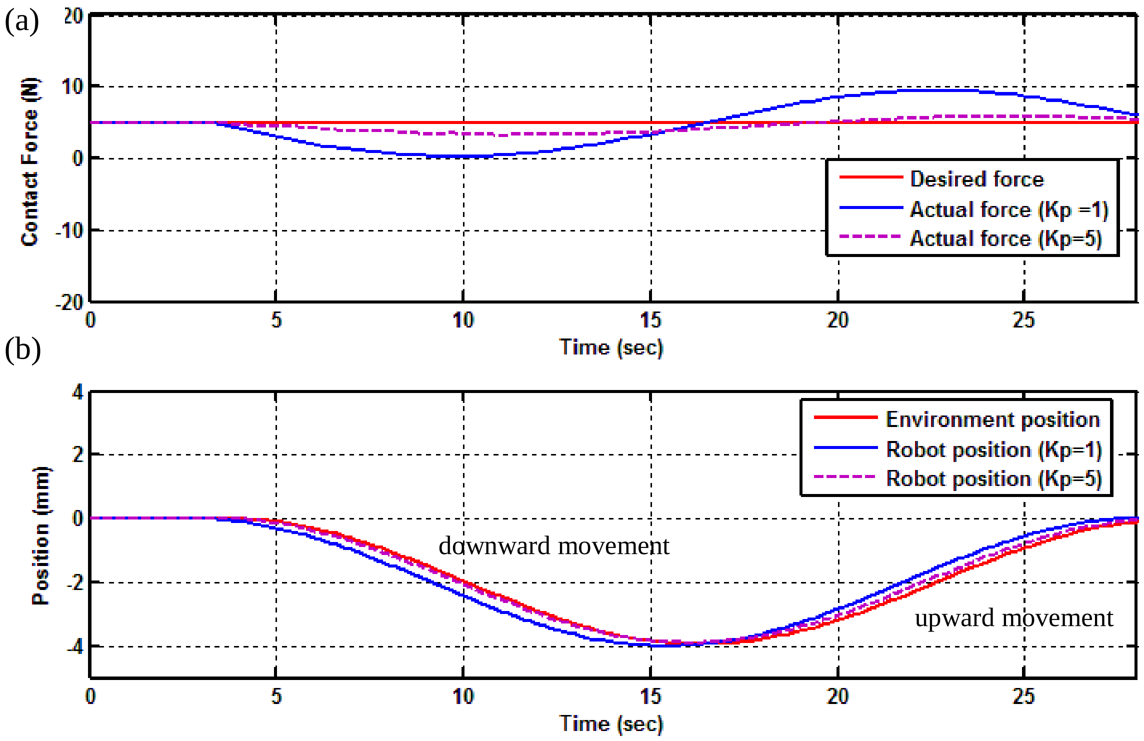


Figure 5-13 a) Contact force response and b) environment and robot position with environment stiffness of 10N/mm and disturbance frequency of 0.25rad/s

To prove the hypothesis, the gain k_p is varied from 1 to 5 and Figure 5-14 illustrates the relationship between the maximum force error and proportional gain for the constant movement frequency of 0.25rad/s. From the graph, it can be observed that the maximum force error is rapidly decreased as the gain increased. The maximum contact force errors during downward movement (away from contact) are higher than upward movement (toward contact). Further discussion can be found in Chapter 8.

Briefly, a higher gain serves to counteract the steady-state error caused by the environment movement. This improved disturbance rejection is certainly an important advantage to be gained from high proportional controller gains. However, choosing the optimum gain for k_p is a cautious exercise. If k_p is too large, the system has high positive force feedback results in oscillatory reaction forces which could easily damage the robot arm, sensor, coil and environment. Figure 5-15 shows the force response beyond k_p = 5 are considered marginally unstable as the force response is poorly oscillated.

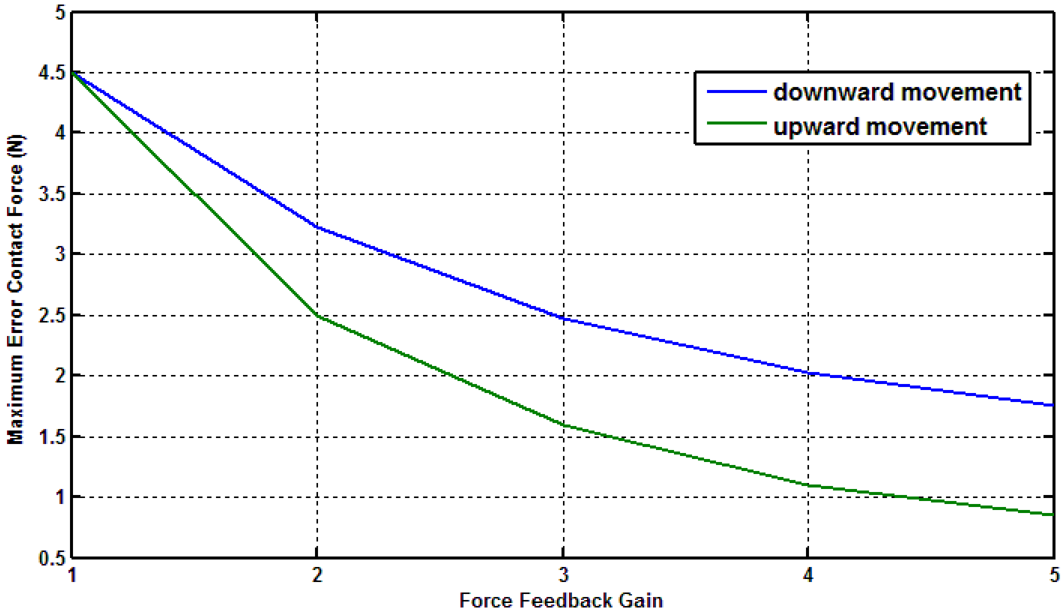


Figure 5-14 Maximum error contact force with varying force feedback gain

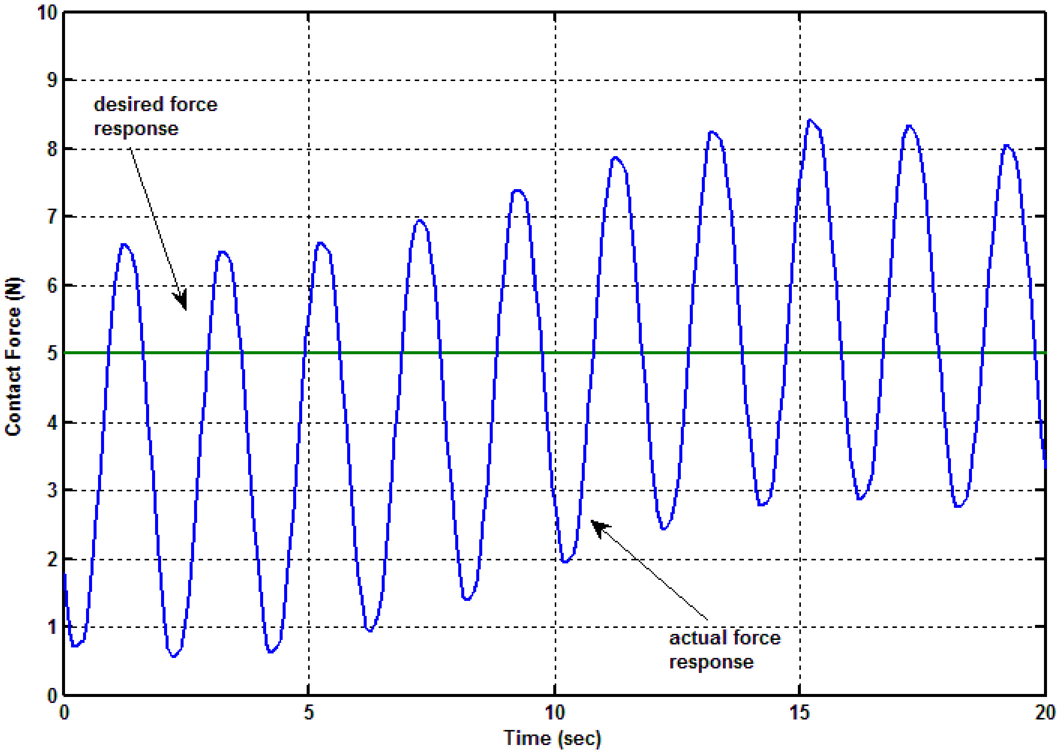


Figure 5-15 Contact force response with environment stiffness of 10N/mm, disturbance frequency of 0.25 rad/s and k_p equal to 6.

The test is repeated with much higher environment position disturbance frequency of 1 rad/s. All other control parameters including environment and desired contact force response are identical to the test response in Figure 5-13. Thus a direct comparison between Figure 5-13 and Figure 5-16 can be drawn, illustrating the effects of the different disturbance frequency. Figure 5-16 shows a relatively similar behaviour with lower ω_d. However, a significant large steady-state error of 97% was recorded. The dramatic effect of the force response variation could be attributed to the low value of robot speed compared to the environment movement as indicated in Figure 5-16(b). This latter result confirms that when the positional disturbance is present, the force transient response is highly affected. The inadequate force response is a result of the disturbance frequency which is larger than system bandwidth frequency.

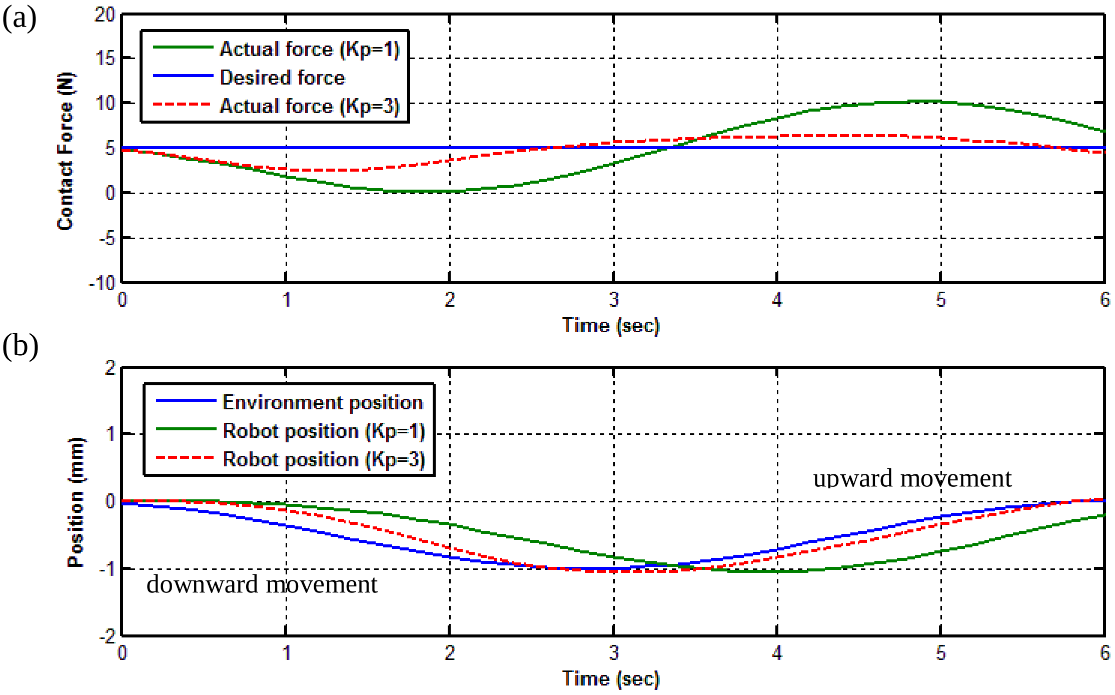


Figure 5-16 a) Contact force response and b) environment and robot position with environment stiffness of 10N/mm, disturbance frequency of 1 rad/s and varying k_p of 1 and 3.

Increasing the force feedback gain further to k_p = 3 agrees with the previous results in Figure 5-14(b) which improved the force transient response steady-state error more than 50% as shown in Figure 5-16(a). Therefore, it can be observed that the robot is closely tracked the environment position as given in Figure 5-16(b), confirming the advantage of increasing the proportional force controller gain.

5.6 Summary

The stability analysis of the external hybrid force control method was performed using an 'idealized' single axis model of the TMS robot system. Two analyses were conducted. The first analysis accessed the system sensitivity to parameter variation namely proportional force controller gain, k_p and environment stiffness, k_e with the aid of Bode diagram and step response analysis. Increasing the force controller gain results in a faster response, but reduces the phase and gain margins which in turn decrease system stability. The system results in sluggish response to a higher value of environment stiffness and overdamped response for a lower value of k_e. The phase margin remains fairly unchanged for different environment stiffness, whilst the gain margin is decreased as the k_e increased, which proved the system tends to instability. It was also observed that the specified value force controller gain must be tuned to specified environment stiffness. These problems are addressed in more detail in force control strategy chapter.

The effect of position disturbance on the system performance has also been discussed. The positional disturbance can be seen to have two main effects; 1) on transient response and steady state errors and 2) on system stability. Unwanted position disturbance interferes with the force control, result in a deterioration of steady-state error and force response as an example as shown in Figure 5-15(a) and Figure 5-16(a). Clearly, in the results described, the positional disturbance has a substantial effect. This judgement depends on the relative magnitude of disturbance frequency, bandwidth frequency, environment stiffness and force controller gain.

At this stage, there are significant problems with the unknown and uncertainties of environment stiffness. A clear understanding of this properties and how to avoid the consequences will be necessary in this work. Clearly, the dynamics of the force-controlled system are now considered to be extremely complex with many independent parameters. Thus, the force control task of maintaining force response as environment stiffness and movement varies with the simple force control law is impossible.

CHAPTER 6

EXTERNAL FORCE FEEDBACK CONTROL STRATEGY

6.1 Introduction

The model of a one degree of freedom (DOF) force-controlled TMS robotic system, the stability analysis of the force control strategy and possible sources of instability have been identified in Chapter 5. Using the developed model, the external force control strategy has been analyzed and confirmed by simulation results, where possible the previous analysis will be compared with reported experimental results later in this chapter.

The external force control strategy design concept will be extended to a 6 DOF robot system implementation. As discussed in Chapter 3, the TMS procedure is divided into two successive tasks. The first one consists of moving the TMS coil to the target position (preoperative phase) and the second one maintaining a low contact force during the treatment (intraoperative phase). In terms of force control strategy, the force control loop can be divided into two conditions;

- Force closed loop control at zero in unconstrained directions
- Force closed loop control at desired force level in constrained directions

Prior to the implementation of robot force control to perform these tasks, the theory behind the external force control has to be addressed in detail. The selection for this scheme depends on two conditions which are imposed for real robotic system implementation; a force/torque sensor must be used to provide force information and the original industrial robot controller must be used. Finally, the remainder of this chapter describes the experimental programme, stability analysis as well as discussion on practical implementation of robot force control.

6.2 Force Control Strategy

6.2.1 External force control structure

The external structure is referring to an external force control loop with an inner position control loop, and force error can be seen as displacement to be performed by the robot manipulator. For better understanding, consider the diagram in Figure 6-1. An incremental position ΔX_F is calculated by force control law to modify the initial position set point X_d. In other words, the operational robot position is controlled to follow a set point which is automatically corrected by closed loop force control so that the force control can be implemented for previous mentioned conditions.

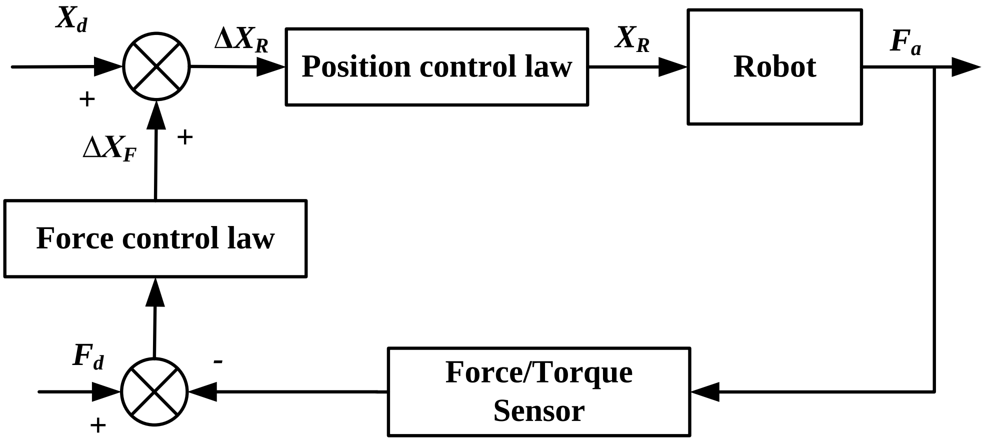


Figure 6-1 the external force control

The main advantage of controlling force using an external control loop, whilst still maintaining an inner position loop, is the relative ease of implementation since it does not require modification to the existing robot control system, as the force control law is defined in Cartesian coordinate and it is possible to input the position control law (original robot controller) with such a Cartesian position/orientation. This is an important feature of the robot control system as the ALTER function for real-time path modification only supports position control in the Cartesian space (further details of ALTER function can be found in Chapter 4). For this reason, it is impossible to develop other force control schemes on TMS robotic system such as parallel hybrid force/position which involves a position control in joint space. In addition, simultaneous control of position and force is easy to achieve with force being controlled by modifying the robot position demand.

For instance, consider the case of *tele-operation* mode in which the force control is used to manually position the robot manipulator by measuring the force applied on the robot end-effector (coil handle) by the operator. With a 'pure' parallel hybrid position/force control, if disturbances occur (or purposely applied) along the force-controlled direction, the induced positioning in these directions cannot be corrected. The problem is not because of the force response quality despite of the variations of the operating conditions, but because of a complete absence of compensating effect due to the nature of the control scheme.

This might be resolved by using an additional robot position sensor in joint position control; however this approach will introduce complexity in high cost and computational burdens. It is reasonable to use an external structure approach in which the possibility of controlling the desired value in certain directions as offered by parallel structure is preserved. Whereas, in the same time provides a fully permanent position control of all directions regardless of the task constraints. This clarification shows that the best control selection to use is the external approach for both *tele-operation* and *tracking* modes. The following section will describe in detail the implementation of this approach in TMS robotic system.

6.2.2 Implementation of an External Force Feedback Control

Figure 6-2 illustrates the block diagram of the force control architecture as implemented in the TMS robotic system. Both internal position and external force control loops are performed in Cartesian coordinates. The key idea of this control strategy is that the force control loop generates the corrective position data from the force error, $e_f\ (t_k)$ through the Proportional-Integral (PI) force controller. The initial contact position $x_d\ (t_{k-1})$ is then modified by the force controller output $\Delta x_f\,(t_k)$. The resultant position demand $\Delta x_d\ (t_k)$ is forwarded to robot controller to determine the next desired position of the robot at every path update interval.

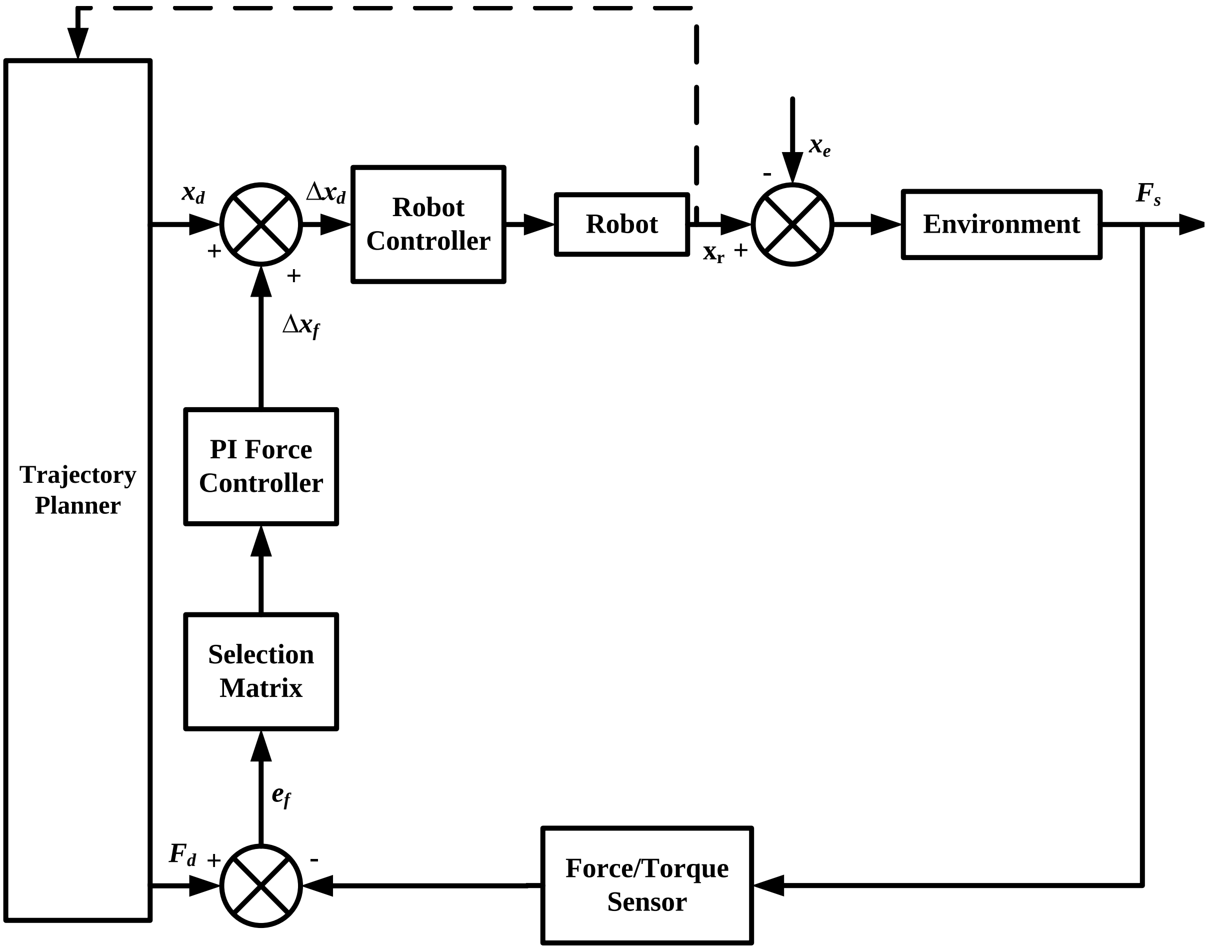


Figure 6-2 Force control strategy based on external force feedback loop closed around an internal position control loop [De Schutter and Van Brussel, 1988b]

When the robot is not in contact with environment (*tele-operation* mode), unlike the De Schutter approach [De Schutter and Van Brussel, 1988a], the desired force F_d is set to zero rather than a constant force. Generally it is unnecessary to apply selection matrix as the axis that naturally constrained in position and force is determined by the input set points x_d and F_d [Perdereau and Drouin, 1993]. However, in the TMS case if the coil position is not vertically aligned with the subject's head, the tangential force due to frictions will affect the position control loop subsequently jeopardizes the desired task. For this reason, the selection matrix is needed so that the closed control loop is correctly separated, i.e. the desired position control direction is not altering by sensed force measurement.

6.2.3 External PI force controller

The force controller is a PI (proportional plus integral) algorithm which essentially determines the amount of correction for a corresponding magnitude of force error, e_f. The proportional value determines the speed of response of the force feedback system. To achieve faster response, the proportional gain can be increased but also results in greater oscillatory behaviour and steady-state error [Visioli, 2006]. The integral term is added to eliminate/null the steady-state error and allows a capability for the controller to maintain a desired constant contact force, F_d. The PI control has been widely implemented in robot force control over the past 30 years [De Schutter and Van Brussel, 1988b; Volpe and Khosla, 1993; Seraji et al., 1996; Zeng and Hemami, 1997; Bigras et al., 2007]. Volpe and Khosla [1993] concludes that the force trajectory tracking is best accomplished with PI control law as it increases the stability of the control system. This view is supported by Seraji et al. [1996] who identified two inherent features of integral term (steady-state tracking setpoint and disturbance rejection characteristics) in their studies and suggest that it is vital component in any practical explicit force control scheme.

The conventional PI algorithm in continuous time expression is described as Equation (6-1).

$$x_d(t) = K_p e_f(t) + \frac{K_p}{T_i} \int e_f(t)\,\mathrm{dt} \tag{6-1}$$

where x_r, K_p and Ti are the demanded robot position, proportional gain and integral time respectively. The digital implementation of the PI controller is based on incremental algorithm form as derived in following equations. This algorithm is preferable since it provides a good performance when applied in practical situations due to fact that the derivative term is sensitive to noise [Stankovic, 1988]. The term is initially included in the force controller but was found to have a destabilizing effect on the system. Given that the overshoot response can be controlled by set up an appropriate proportional gain, the derivative term has since been removed. To derive the discrete-time of PI controller, both side of equation (6-1) is differentiated as given in following equation (6-2) [Li and Gatland, 1995; Aström, 2002; Visioli, 2006].

$$\dot{x_r}(t) = K_p \dot{e_f}(t) + \frac{K_p}{T_i} e_f(t) \tag{6-2}$$

Based on Aström's discretization method [Aström, 2002; Visioli, 2006], the algorithm is derived by using backward finite differences as follows

$$\frac{x_r(t_k) - x_r(t_{k-1})}{T} = K_p \frac{e_f(t_k) - e_f(t_{k-1})}{T} + \frac{K_p}{T_i} e_f(t_k) \tag{6-3}$$

where T denotes the sampling time.

The above equation (6-3) can be rewritten as

$$x_r(t_k) = x_r(t_{k-1}) + K_p \cdot de_f(t_k) + K_i \cdot e_f(t_k) \tag{6-4}$$

where $K_i = \frac{K_p}{T_i} T$ is the integral gain. The process of experimentally selecting the controller gain parameters to meet desired performance specifications is known as controller tuning. For the force controller, it is critical that the overshoot response is kept to a minimum but at the same time, the response of the force controller should be fast as possible. This tuning of the controller can be tedious process and often involves iterative process that requires the operator's experience and involves trial and error process until the desired specification is required. Thus, a systematic approach based on the second method of Ziegler and Nichols rules [Ogata, 2009] is applied to simplify the determination of controller parameters.

6.3 Stability issues in Robot Force Control

Stability and robust performance are important issues in implementing robot force control. Many researchers have carried out extensive stability analyses and various explanations for contact instability have been discussed. Numerous research studies of the stable force control has proven to be quite difficult upon contact with a rigid surface resulting from an implementation having low-bandwidth of the control loop [Hogan, 1988; Burn et al., 2003; Short and Burn, 2007]. A number of approaches have been proposed to improve force control by modifying either the force control strategies or robot design modification. For example, Hogan [Hogan, 1985a; Hogan, 1985b] introduced impedance control to control the robot/environment contact force without the need for force feedback. This control method employs the endpoint impedance which has a dynamic relationship between the environment motion and the interaction force of the robot. Other studies have considered implementing passive compliance devices between the robot and environment [De Schutter, 1987; Whitney, 1987]. However, both methods introduce some limitations in the force control strategy which are cleared unsatisfactory in TMS application due to lack of knowledge on subject's head characteristics which can be assumed as unknown or uncertain contact environment.

An uncertain contact environment is one of the major problems in the implementation of robust force control. The essential stability trade-off between force feedback gain and environment stiffness has been identified many times [Eppinger and Seering, 1987a; Eppinger and Seering, 1987b; Kazerooni, 1987; Whitney, 1987; Ow, 1997; Burn et al., 2003]. Burn and Short [Burn et al., 2003] demonstrated that a problem arises when the environment stiffness is unknown and variable since a fixed-gain controller gain is defined for a specific value of environment stiffness in order to achieve desired requirement. Ideally, the force controller gains should be adjusted to reflect the environment stiffness.

The approach begins with initial force control attempts; from this, comparisons with the previous analysis are drawn which attempt to optimize the performance of the system. During the development of the force control algorithm, an empirical study was carried out to study the influence of force feedback gain and the environment stiffness on the system stability. These problems have already been introduced in the Chapter 5, and will now be considered experimentally. The following section describes the tests conducted including the validity of the previous analytical model. The experimental test has been conducted using the designed TMS robotic system as described in Chapter 4. Judgements to the suitability of the force control are made by considering transients of the force response. The response was recorded to illustrate the salient features of proposed robot force control.

6.3.1 Effect of environment stiffness and force feedback gain variations

6.3.1.1 *Test description*

A cantilever beam contact test was used to illustrate the effect of different environment stiffness on the force control stability [Ow, 1997; Po-ngaen, 2006]. Figure 6-3 shows the cantilever beam contact test set up schematic diagram.

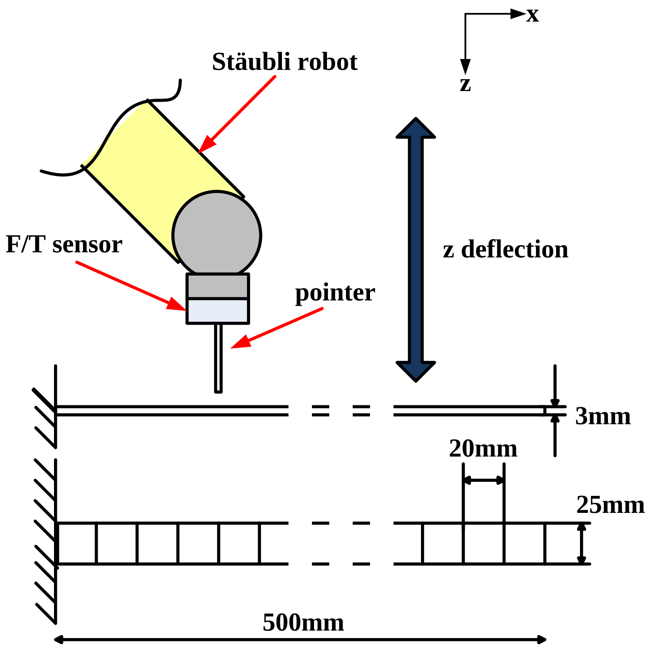


Figure 6-3 The cantilever beam experiment

A marginal gain test was conducted in which a force step input was executed and the proportional gain of the force controller (in this case, only the P controller is used) was varied until the system exibited marginal stability ie. when the system first exhibit sustained oscillations.

Prior to the experiment, the beam stiffness was predetermined by measuring the load-deflection response. A slender 500mm x 25mm x 3mm steel flexible beam is rigidly clamped at one end, mounted horizontally and parallel to robot's *x* axis. A dial gauge with 0.01mm resolution was used to measure the beam deflection when a 10N force was applied vertically at each 20mm gradual mark. The theoretical beam stiffness k was calculated by using elastic beam theory as shown in following equation;

$$k = \frac{3EI}{L^3} \tag{6-5}$$

where E is the Young’s Modulus of steel, I is the moment of inertia of the beam cross section and L is the length of the beam. The beam stiffness results obtained from both methods are presented in Figure 6-4. As can been seen from the graph, there was a significant positive correlation as experimental results show a similar trend with the theoretical values. However, there is discrepancy at displacement lower than 100mm from a fixed end point due to clamping of fixed end. Albeit the error, the experimental results are acceptable since the values lie between approximate measurement probability intervals.

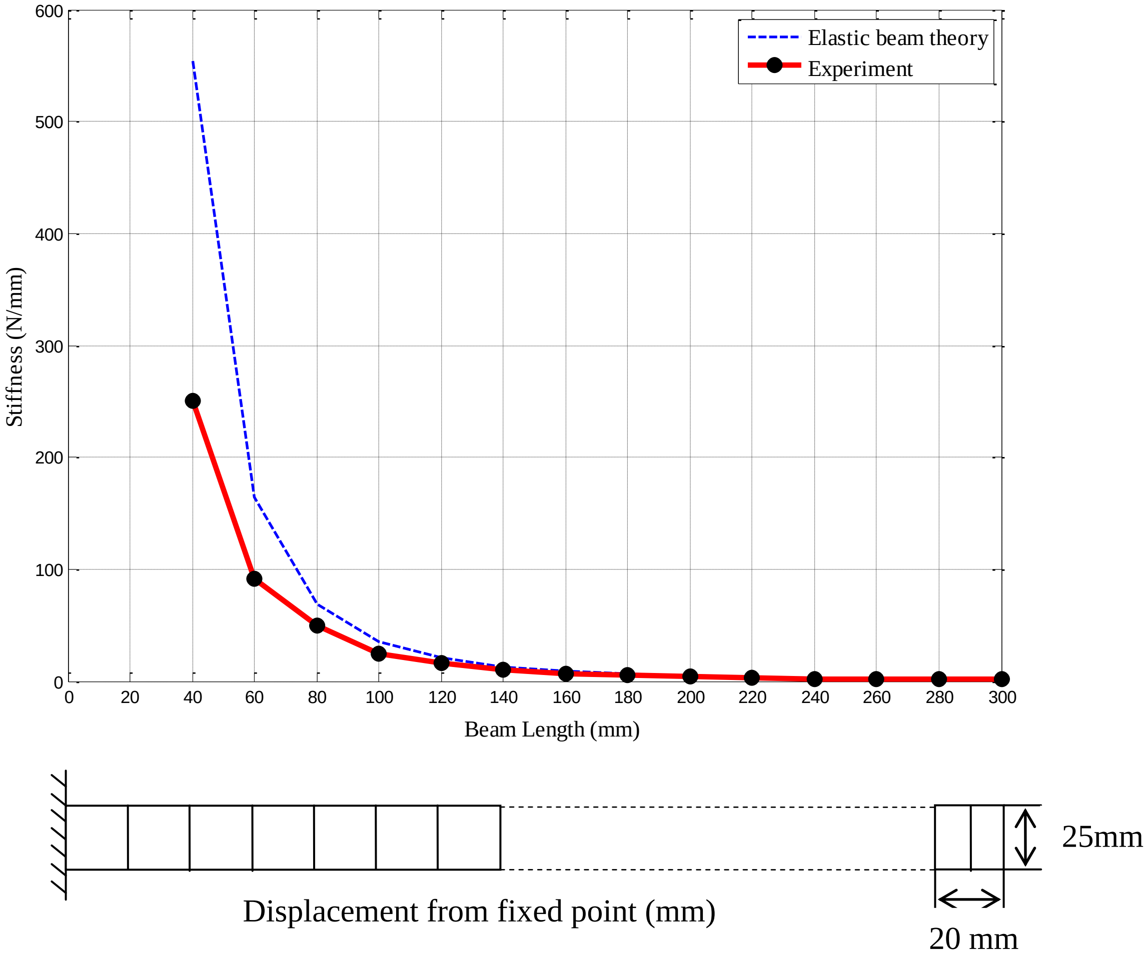


Figure 6-4 Environment stiffness values for different flexible beam displacement

6.3.1.2 *Results and Discussion*

Figure 6-4 indicates how the measured cantilever stiffness varies according to its length. A number of marginal gain tests were carried out for different predetermined cantilever stiffness and the compilation results are shown in Figure 6-5. As expected, the graph indicates the control gain is inversely proportional to the environment stiffness. This result is consistent with analysis in Chapter 5 and those of other studies [Ow, 1997; Burn et al., 2003; Po-ngaen, 2006] which suggest that the stiffer the environment, the smaller the permissible control gain. In contrast, a higher control gain is applied for a softer environment. The evidence of this result can be clearly seen in the case of soft environment, the compression of the environment's surface δx_e will cause only a small change (δF_d) in the desired contact force F_d. The relative change is proportional to the ratio of the proportional gain K_p to the environment stiffness K_e and is given in equation (6-6).

$$\frac{|\delta F_d|}{|F_d|} = \frac{|\delta x_e|}{|x_e - \delta x_e|} = \frac{K_p}{K_e} \tag{6-6}$$

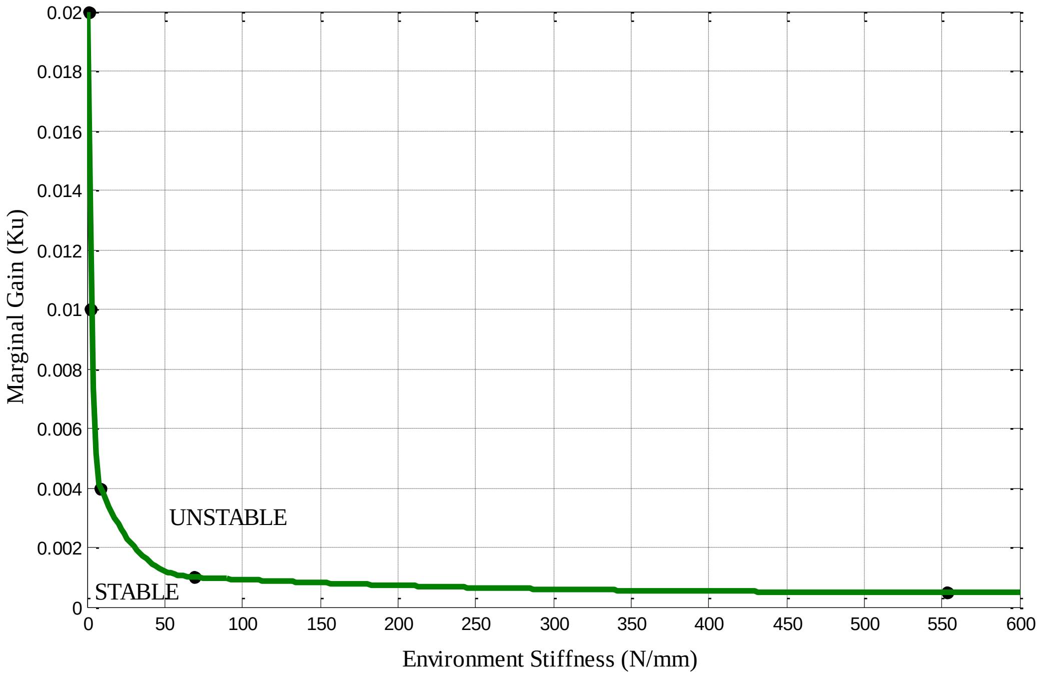


Figure 6-5 Proportional marginal gain K_u versus Environment stiffness K_e

Having identified the proportional gains, the PI force controller has been tuned carefully to the desired performance specification for different environments. Figure 6-6 shows the force step response using the selected gains on the different environments for the implemented force control system. From the graph, it is apparent that the controller response for softer environments give faster rising time compared to stiffer environment, but the settling time are decreased. The result may explain that the fast movement of the robot causes the flexible beam to vibrate and slow to decay thus prolonging the settling time. Table 6-1 provides proposed PI force controller gains for several identified environments. The table presents recommended ideal controller gain values for identified environments.

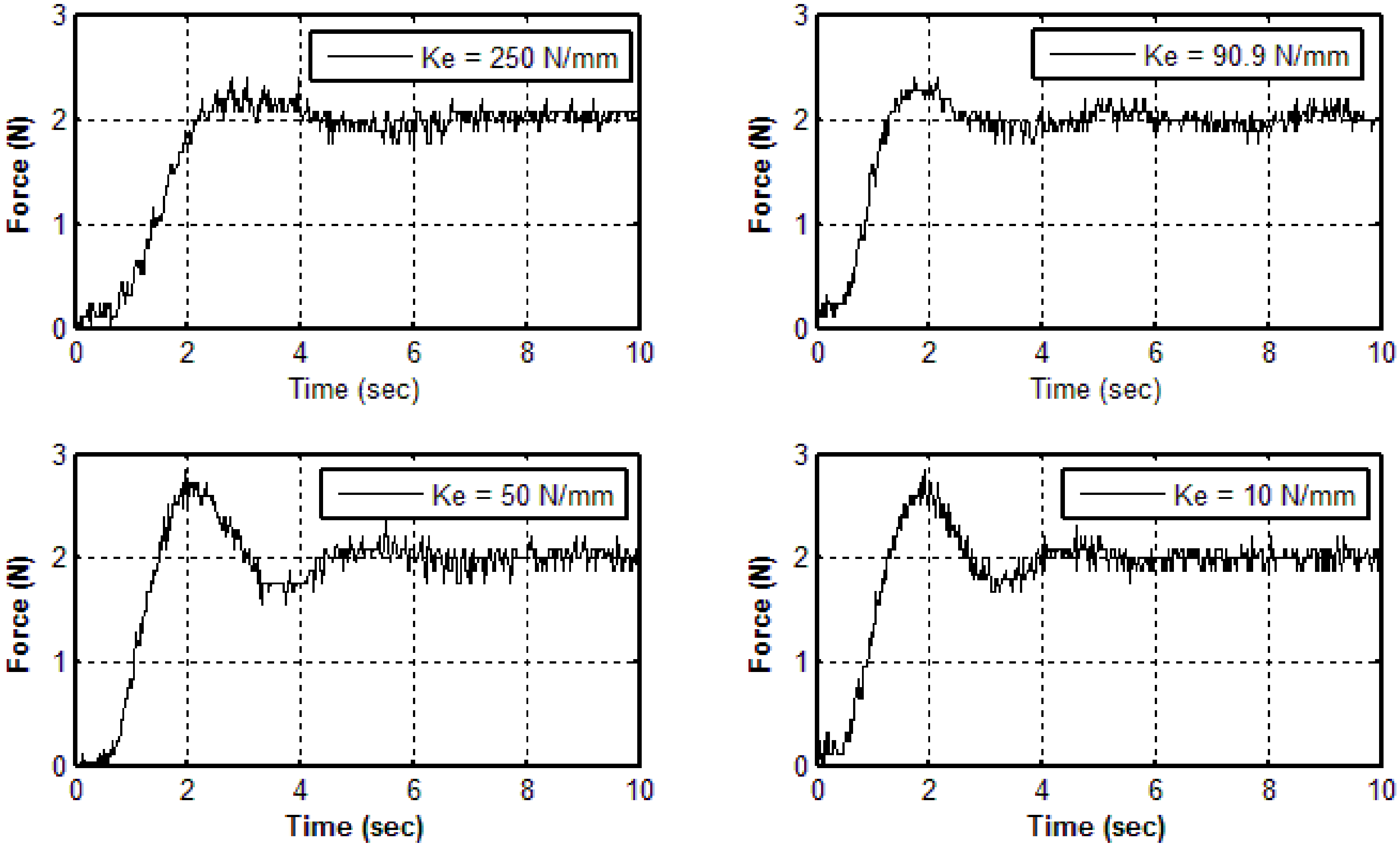


Figure 6-6 Force step response for different environment stiffness

In general, it can be suggested that the stable force control can be achieved if the environment stiffness is known and it would be problematic for the uncertainty environments. The findings enhance an understanding of the role of controller gains in order to achieve desired robust performance.

Table 6-1 Proposed PI controller gains for different environments

Environment Stiffness (N/mm)	**Proportional Gain** *Kp*	**Integral Gain** *Ki*
250.0	0.0004	0.0001
90.9	0.0008	0.00067
50.0	0.0012	0.0005
10.0	0.002	0.0015

In principle, designing a fixed-gain conventional controller to achieve a desired specification for specific environment is relatively straightforward. The flexible beam experiment was repeated to identify the effect of unknown or variable environment stiffness. The robot is commanded to follow a force step input of 2N for three different values of force feedback gain: K_p = 0.01 in Figure 6-7(a), K_p = 0.005 in Figure 6-7(b) and K_p = 0.0001 in Figure 6-7(c). The results of Figure 6-7 allow easy comparisons of variation in proportional force feedback gain for predetermined environment stiffness of approximately 50N/mm. The graphs show that the stability decreases with increasing of proportional force feedback gain K_p. It is worthwhile observing how the transient response is significantly overdamped when the controller is tuned for low proportional gain (stiffer environment) with a relatively high settling time. In contrast, when it is applied to a soft contact (high proportional gain), a large peak overshoot and oscillatory behaviour is observed.

This result is consistent with the simulation result of section 5.3.2 that an increase in proportional gain is accompanied by a decrease in gain and phase margins which indicates the system tends to the instability condition. A further investigation on varying environment stiffness tests explained that K_e is an extremely important variable in force control and cannot be ignored in force control tasks. Chapter 5, involving a one degree of freedom simulations, also confirms that environment stiffness is a vital parameter for robust and stability of force control. The present results are significant in a need to find another technique such as gain scheduler or adaptive force controller to maintain desired force response as the environment stiffness varies or is unknown.

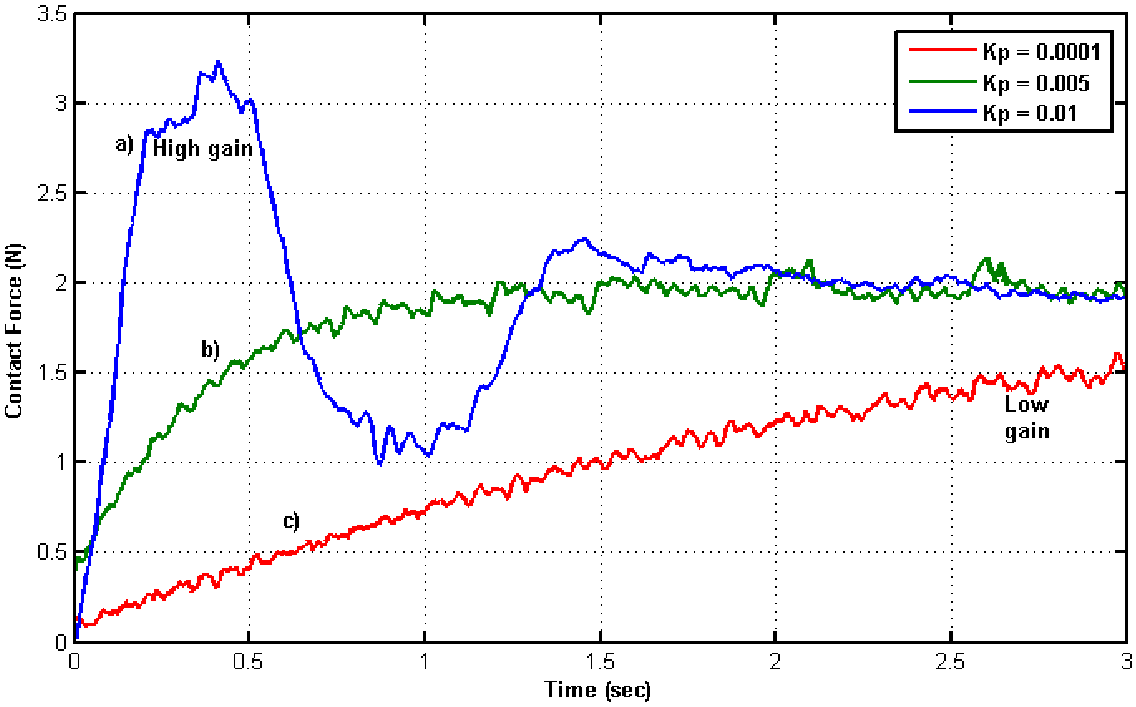


Figure 6-7 Result of effect of varying K_p of conventional force control system

6.3.2 Effect of positional disturbance (moving environment)

6.3.2.1 *Test Descriptions*

Thus far, the stability of the force controller has been tested on fixed environment. Further test will illustrate another problem experience with the force-controlled TMS robotic system through a test designed to investigate the effect of varying environment position. In order to evaluate the force controller capability on moving environment, a ball is chosen as a contact environment as described in section 3.2.1 with K_e approximately 10N/mm as shown in Figure 6-8. The values of proportional (K_p = 0.002) and integral (K_i = 0.0015) gains are set up and found experimentally as provided in Table 6-1.

The test begins at the vertical orientation of the ball in which the control system is set up with the force control in *z* axis with the other 5 axes is set up with position control.

The *z* axis of the Stäubli robot is then moved down at an approximate speed of 10 mm/sec until it makes contact with the ball. Once contact is made, the operator, with the aid of the system information display, activates the autonomous force control mode.

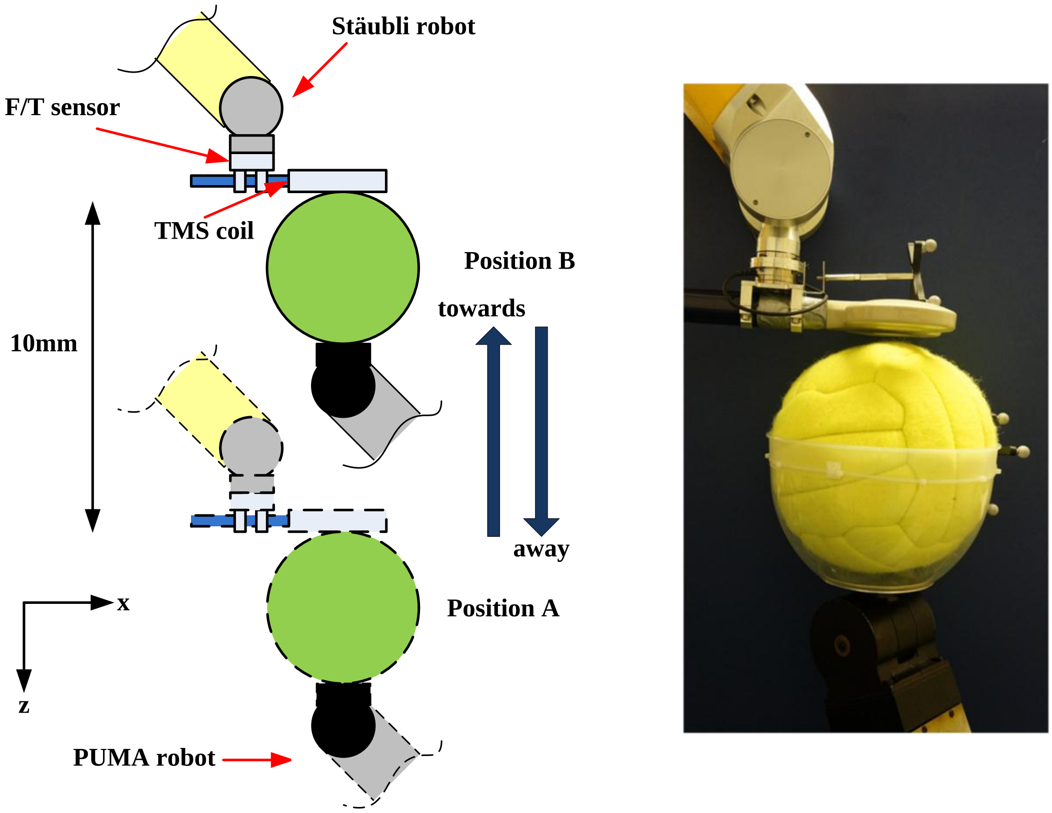


Figure 6-8 Schematic diagram of environment position changing test

To demonstrate the effect of moving environment, the robot is required to maintain a desired contact force of 2N. Once the steady-state value of contact force is reached, the PUMA robot is commanded to move towards (upward direction) the Stäubli robot at low velocity of 1 mm/s. This is followed by 10mm movement away (downward direction) from the Stäubli robot. As can be seen from Figure 6-9, the force response is immediately increased during towards movement, and then is restored rapidly to the desired force setpoint 2N. Subsequently as PUMA robot moves away from the Stäubli robot, the force response is decreased momentarily nearly zero, and then is restored to the desired force setpoint F_d = 2N back after 20 seconds.

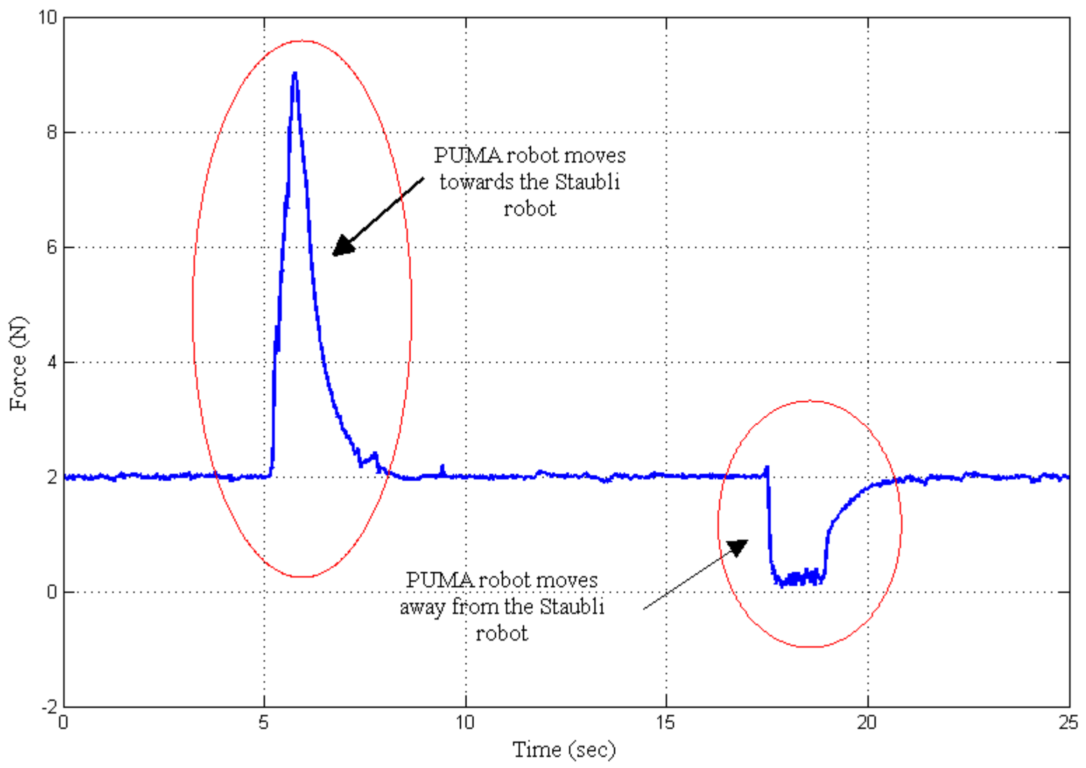


Figure 6-9 Force control (varying environment position)

6.3.2.2 *Results and Discussion*

Several interesting observations are evident, from Figure 6-9, the force control can be seen to retrace the demanded force back for both conditions. However, the response is evidently undesirable in which the force controller response is degraded as it takes about 2~3 seconds to recover back to the desired force. This is more noticeable in the case of downward movement, where the Stäubli robot nearly loses contact with the PUMA 560 robot. This result is most likely owing to the lower speed of Stäubli robot compared to the PUMA robot. This is due to the low value of force error and control gain. In this study, the environment movement direction was also found to vary the stiffness of the environment. Refer to section 8.3.1 for further details. Besides, this can be more challenging as discussed in previous section 6.3.1 in which environment stiffness can limit the high-gain settings that were required for a force controller to compensate the environment movement. To make matters worse, there is a possibility that the environment moves fast and stops suddenly and subsequently initiates a limit cycle. If this does occur, there is strong probability that Stäubli robot would not be able to maintain a constant low contact force on the environment surface and subsequently introduce harm to the environment. For this reason, the force control gain should be tuned for minimal overshoot and it is very important that the force controller ensures the Stäubli robot remains in contact with the environment to avoid any undesirable robot movement and excessive contact force.

6.4 Preliminary test of free space and constrained motion

As discussed previously, the force-controlled TMS procedure is sub-divided into two tasks, namely *tele-operation* mode (free space motion) and *tracking* mode (constrained space motion). The *tele-operation* mode is used to move the robot to the desired contact point and once the robot is in contact with the human head, the *tracking* mode is activated to maintain the desired contact force during the TMS procedure.

6.4.1 Free space test

The preliminary test involves the operation of the robot system in free space motion. Prior to the experiment the robot was initialized by the system program to move to a predefined start position which is essential to provide a comfortable position to the operator to manipulate robot movement. The arm is compliant as the desired force F_d is set to zero and the arm does not move until the direct force is applied on the robot's end-effector. The test is designed to verify the applied force induces a robot motion along the correct axis (i.e. the force is applied along the z axis direction and the robot motion in z axis is observed). A 200g mass was attached to the robot's end-effector to induce a constant force (~2N). The position and force response is observed and shown in Figure 6-10. The proportional gain K_p is set to 0.004 and was found experimentally.

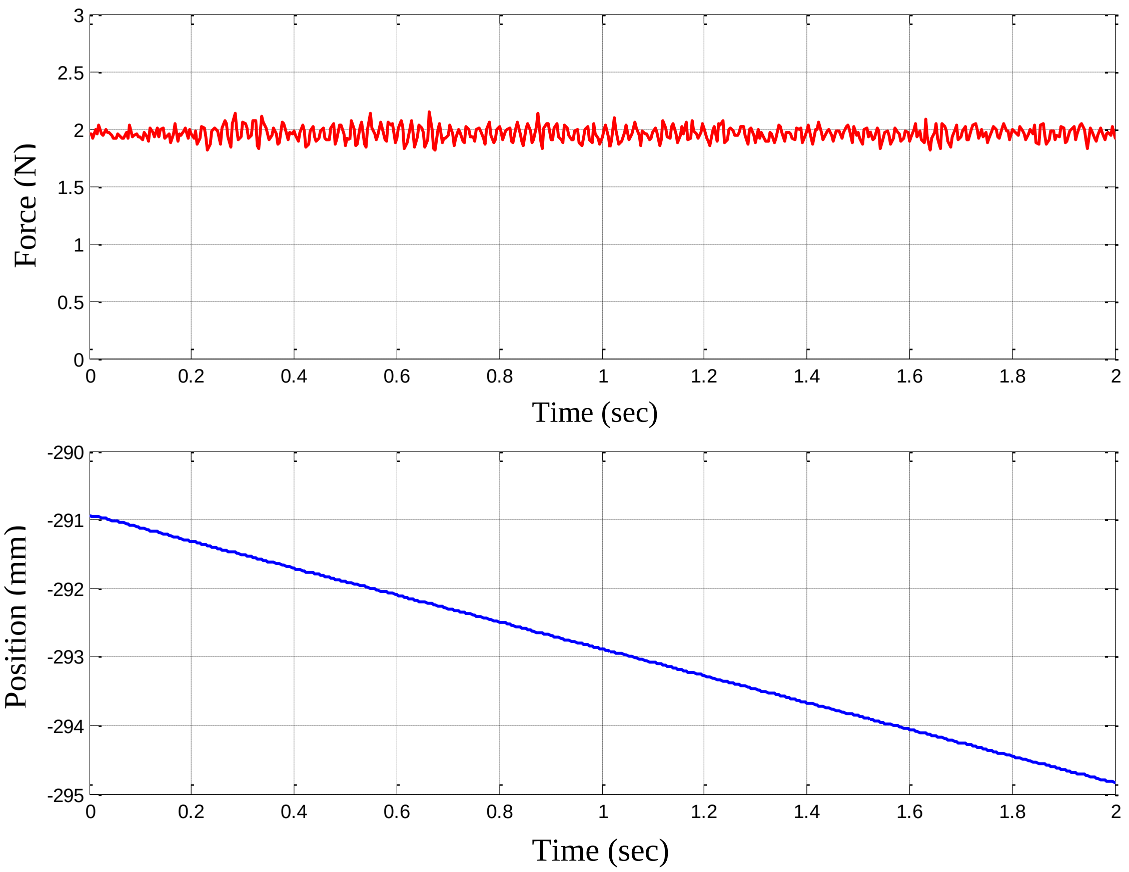


Figure 6-10 Force and position response of a free space test experiment

The block diagram of free space motion is illustrated in Figure 6-11.

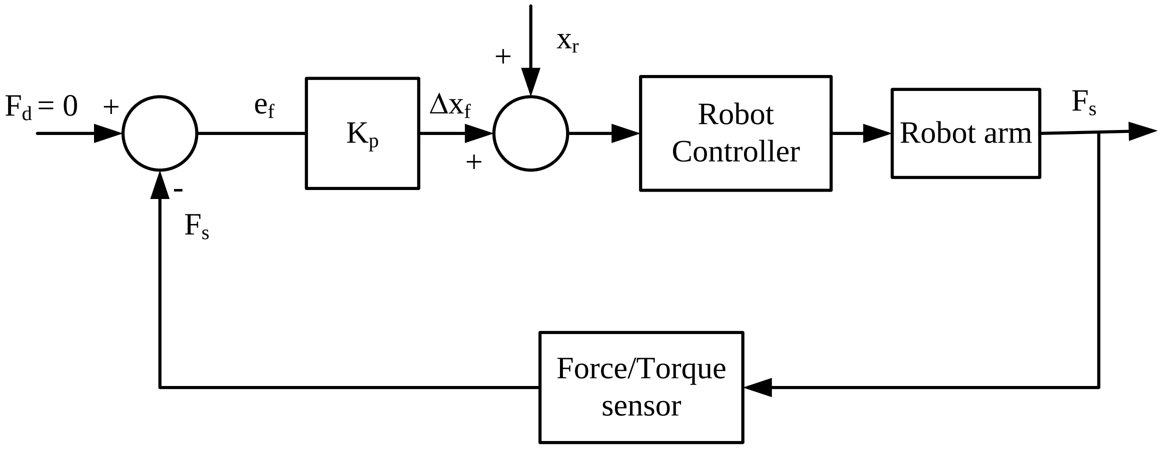


Figure 6-11 Schematic diagram of the free space motion

The sensed force F_s generates incremental position demand Δx_f for the robot system and then the robot arm will move from the current position at velocity proportional to the applied force as shown in following equations.

$$\Delta x_f(t_k) = K_p F_s \tag{6-7}$$

6.4.2 Constrained space test

In the *tracking* mode, the arm is commanded to maintain desired constant force with environment ($F_d \neq 0$N). The PI force control loop compensates the difference or error between the desired and actual force as shown in equation (6-8) and then routed the desired incremental position to the robot controller as shown in Figure 6-12.

$$\Delta x_f(t_k) = K_p \cdot de_f + K_i \cdot e_f \tag{6-8}$$

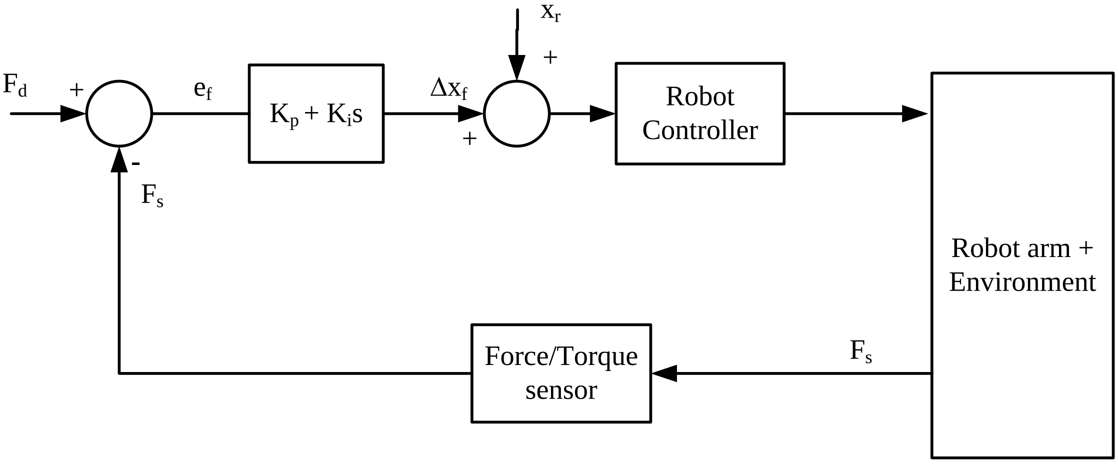


Figure 6-12 Schematic diagram of the constrained space motion

In order to evaluate the force controller capability, the ball is again chosen as a contact environment (as described in section 3.1.2) with K_e approximately 10N/mm, as shown in Figure 6-13. The values of proportional (K_p = 0.002) and integral (K_i = 0.0015) gains are set up and found experimentally. The test is design to simulate a variety of conditions namely a sequence of step inputs (2N, 4N and 1N).

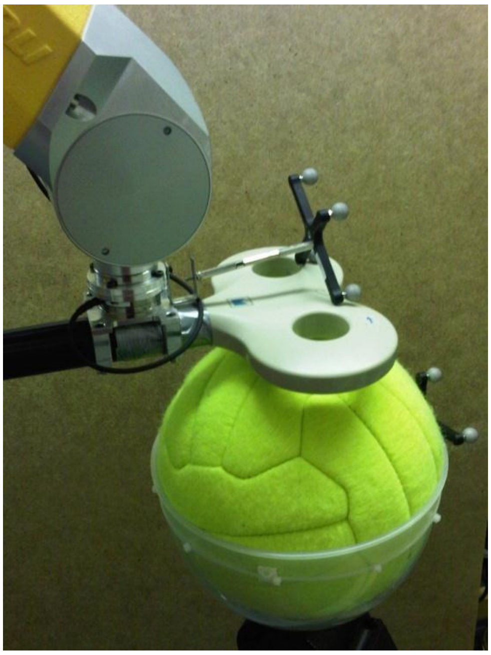


Figure 6-13 Constrained space test

The test begins at the vertical orientation of the ball in which the control system is set up with the *z* axis force control with the other 5 axes are position control. The *z* axis of the Stäubli is then moved down at an approximate speed of 10mm/sec until it makes contact with the ball. Once contact is made, the operator, with the aid of the system information display, activates the autonomous force control mode.

The system initially is driven by the desired force of 2N and was then held constant for 8 seconds before a step change of 4N and 2N. Finally, the step change is reduced to the contact force demand back to 1N. During this force cycle, the PUMA robot is maintained in a fixed configuration. Figure 6-14 provides the results obtained from the test.

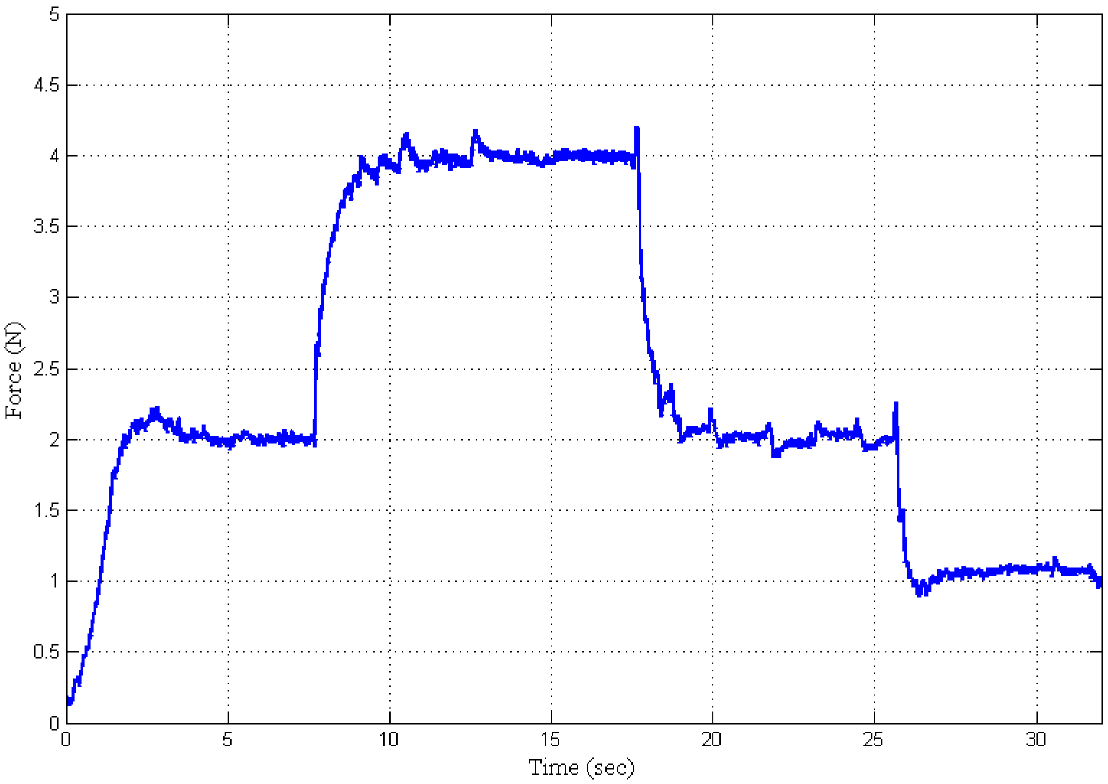


Figure 6-14 Force response to step input

It is apparent from Figure 6-14 that the contact force is shown to track very closely the demanded input with approximately zero steady-state error in all conditions. During the initial step period when the robot was moved towards the ball, a minimal 15% overshoot is observed. The results also demonstrate the transient response is slightly oscillated but are damped out rapidly.

6.5 Discussion

6.5.1 Force signal noise

The force signal measured by the ATI force/torque sensor normally contains a noise component which has a detrimental effect when the control loop has any differentiating components. For this reason, most researchers implemented PI controller in force control loop [Raibert and Craig, 1981; De Schutter and Van Brussel, 1988b; Volpe and Khosla, 1993; Degoulange and Dauchez, 1994; Seraji et al., 1996; Zeng and Hemami, 1997; Bigras et al., 2007]. To overcome this problem, a first-order-filter is often employed to remove the noise, however it inevitably introduces an additional phase-lag into the system which reduces the stability margin and contributes instability in closed-loop control [Seraji et al., 1996; Ow, 1997]. A further stability analysis must be carried out if the filter is added to the control system. It was found that the integral gain in force control algorithm significantly eliminates the steady-state error which acts in sense of a low pass filter and attenuates the force measurement noise at its output.

6.5.2 Effect of K_p and K_i in force control loop

Figure 6-7 shows that the proportional gain determines the force response time. However, if the K_p is too large, the overshoot is increased and prolonged the settling time. The proportional controller always reduces the error but does not eliminate it. Thus, the integral gain is introduced to correct any error (offset) that may occur between the desired force and actual force. Once the proportional gains were established, the PI force controller was tuned carefully to achieve the desired performance specification. Figure 6-15 illustrates the effects of adding integral gains. It is found that increasing K_i will decrease the tendency for oscillations (reduced overshoot), increase the settling time as well as eliminates steady-state error. The wind up phenomenon introduces by the integral action in control loop is automatically avoided due to implementation of the incremental PI algorithm in force control loop.

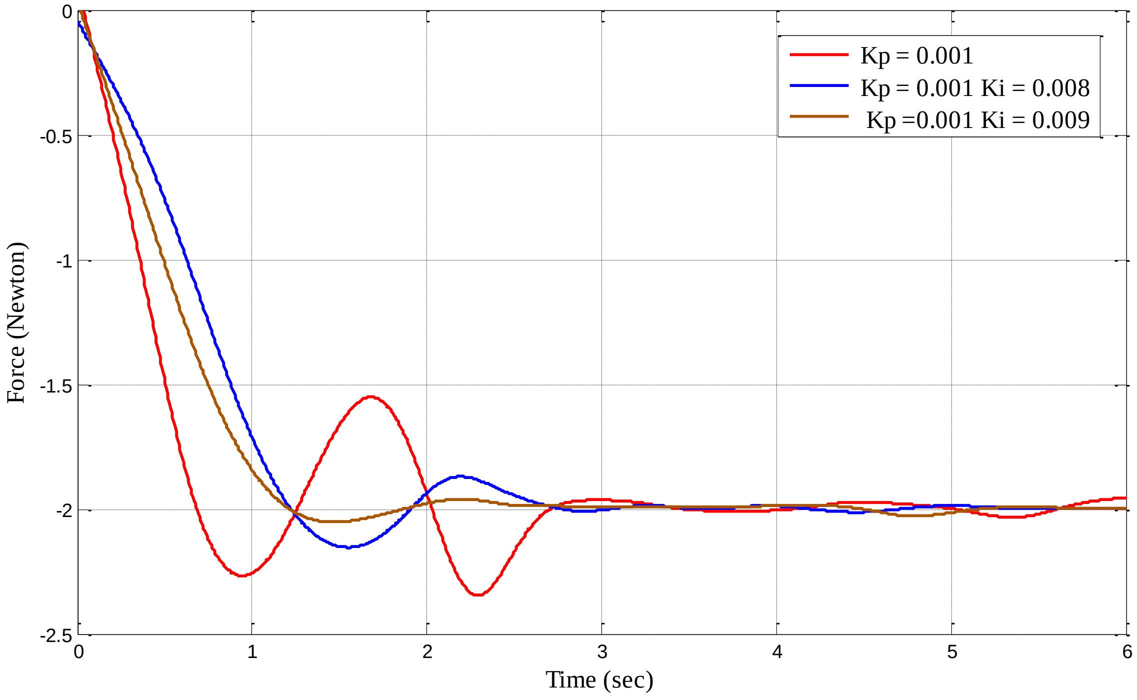


Figure 6-15 Effect of adding and varying Ki

6.5.3 Properties of external force control

Most researchers pay no or little attention to the transition control between unconstrained and constrained motion. Normally, the compliant motion with force feedback is assumed as physical contact with environment, thus many studies considered the free space movement is purely position control. Experiment results proved that the external force control is successfully applied in both mentioned motions with acceptable overshoot (10 to 20%). Furthermore, the control approach is proved to ensure a force tracking capabilities which is vital to TMS robotic system. Finally, it is noted that the control approach presented is pragmatic since it is easily implementable on industrial robot controller where access to the joint torques is often not provided.

6.5.4 Environment stiffness and moving environment

The findings enhance understanding of the role of controller gains in order to achieve desired robust performance. In general, it can be suggested that the stable force control can be achieved if the environment stiffness is known and it would be problematic for the uncertainty environments. This study is consistent with those described by Ow [1997] who found that environmental uncertainty is a major problem in the implementation of robustness practical controllers. Ow's environment stiffness detection technique showed accurate performance and smooth switching between controller gains. However, this slows down task execution and can results in unstable contact when the environment stiffness varies significantly. Recently, several intelligent control techniques such as fuzzy logic, neural network and genetic algorithms have been published to improve the robot performance with various contact situations. The intelligent approaches have the ability to incorporate decision making and heuristics into the complex system controller design. The other advantages are it is potentially very robust and highly effective in maintaining a good closed loop system performance over a varied range of operating conditions. The following chapter will give an overview of intelligent control techniques as well as its implementation to the robot force control to solve identified problems.

6.6 Summary

In this chapter, the external force control strategy has been verified by experimental evaluation. A series of tests were conducted to demonstrate the effectiveness of the force-controlled TMS robotic system. An external force control strategy which employs an external force feedback loop in conjunction with the existing inner robot position controller was implemented. The force controller is a PI force controller in which the gain was tuned to the performance specification. Prior to programming the robot to perform the TMS operation tasks, the issue relating to force control stability was first addressed. The implementation of a conventional force controller in TMS robotic system was found to be difficult. The preliminary test result suggests that unknown or uncertain environment stiffness and positional disturbance would drive the system into undesired behaviour. Thus, an intelligent approach is proposed to overcome these identified problems as discussed in the following chapter.

CHAPTER 7

INTELLIGENT APPROACH TO ROBOT FORCE CONTROL

Fuzzy logic and neural network systems have attracted renewed interest from researchers in a variety of scientific and engineering disciplines. Theoretically, these system fundamentals and inspiration are very different but there are a number of parallels that point out similarities. Fuzzy logic is based on how the brain deals with inexact information while a neural network is modeled as physical brain architecture. Both fuzzy logic and neural network systems have been shown to have the capability of modeling complex nonlinear processes to arbitrary degrees of accuracy.

A brief review of the motivation and technique behind each of these computing approaches will be discussed in sections 7.2 and 7.3. Prior to Neuro-Fuzzy implementation on the force controller, the Proportional-Integral (PI) Gain Scheduling was used in attempt to improve the conventional controller. This approach was applied via a combination of fuzzy-like IF-THEN rules and PI controller to characterize the controller parameters with a desired behaviour.

Following the review, the adaptive Neuro-Fuzzy method design will be described, which combines the advantages of both fuzzy logic and neural networks. The remainder of this chapter is devoted to the preliminary test programme that provides a comparison of dynamic response of the system using a conventional method PI, PI Gain Scheduling and the Neuro-Fuzzy approaches.

7.1 Motivation behind Intelligent Approach to Robot Force Controller

The previous chapter has described how the Stäubli robot can be preprogrammed to perform force control on defined environment stiffness and fixed configuration. However, in TMS application the subject’s head movement is unconstrained in which the contact environment can be assumed as unknown. Furthermore, the subject’s head are more complex and the stiffness is also varied from one person to another. In such situation, trying to program the robot to perform the exactly same procedure force control can be self-defeating as the force controller need to anticipate all the corrective actions for all possible situations which might arise during the repeated TMS procedure. In these circumstances, an alternative strategy is to design the force controller based on human-decision making algorithm to carry out force control on such an unknown environment.

7.2 Fuzzy Logic Control

The idea of Fuzzy Logic was introduced by Zadeh in the mid-1960s [Ross, 2004] and became an important asset in control engineering. Modern control systems can be extremely complicated as the mathematical model is rarely accurate and the existing nonlinearities always influence the behaviour of the system. Furthermore, there is always a difficulty modeling disturbances that also influence the system behaviour. Thus, the Fuzzy Logic theory that is based on human decision-making is ideally suited to overcome this problem since it is robust and can control a nonlinear system. By designing a controller based on a set of IF-THEN rules, several control applications including washing machines, air conditioners, industrial robots, and many other applications can take advantage of the inherent features of fuzzy logic control.

The pioneer of Fuzzy Logic theory in control applications is Mamdani [1974] who developed Fuzzy Logic algorithm to control a steam engine. The set of fuzzy rules is derived from experienced human operators to use under Fuzzy Logic to form a control policy. There are many books on the mathematics of fuzzy sets, fuzzy logic and fuzzy systems for further discussions; see the references by Jantzen [2000], Ross [2004], Passino and Yurkovich [1998], Yen and Langari [1999] and Driankov et al. [1996].

7.2.1 Structure of Fuzzy Logic Control

A block diagram of a fuzzy control system is shown in Figure 7-1, where the fuzzy controller is composed of the following elements;

- A fuzzification interface
- A knowledge base
- An inference mechanism
- A defuzzification interface

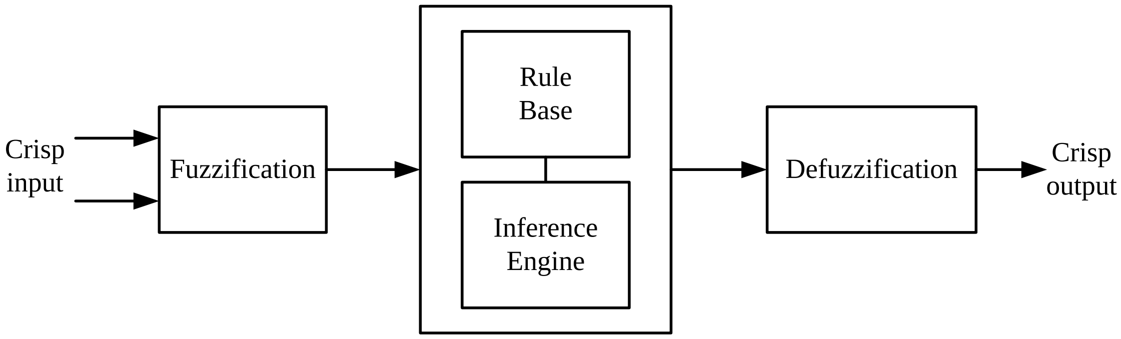


Figure 7-1 Basic structure of fuzzy logic control [Reznik, 1997; Passino and Yurkovich, 1998; Ross, 2004; Jantzen, 2007]

The fuzzification interface is a subjective evaluation process that maps all the possible control inputs to the corresponding universe of discourse fuzzy sets. The fuzzy sets are described as linguistic terms of the input variables and are represented as membership functions (MFs), assigned by an expert. For example, assume the next fuzzy set in Figure 7-2.

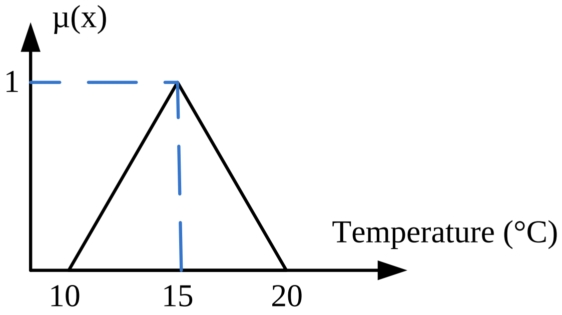


Figure 7-2 Fuzzy set 'Cold'

The temperature 11° is assigned to a value of 0.1 for the fuzzy set ‘Cold’. The number of the sets and the choice of their membership functions depend on the control application. The choice of fuzzy variables may have a substantial influence on the controller behaviour. However, there is no unique solution or design procedure that can be followed to obtain either the most effective membership function types or their number. Increasing the number of membership function can enhance the controller behaviour; however this will consequently introduce a computational time burden. The fuzzification process should be optimized based on the knowledge about the control system, which help in identification of the suitable membership functions, linguistic terms and their ranges to suit the system’s behaviour. The fuzzy sets may overlap to allow smooth transition between the states of the system or may be completely separate and may have different shapes as shown in following Figure 7-3. Thus, it can be said that the previous temperature 11° belongs to the fuzzy set ‘Cold’ for 10% and to the fuzzy set ‘Very cold’ for 80%.

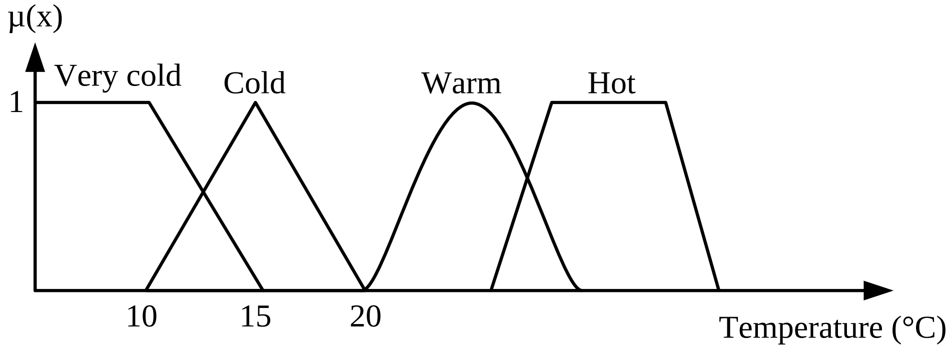


Figure 7-3 Fuzzy set ‘Cold’

The knowledge base comprises a set of linguistic rules that capture the expert’s knowledge about how to control the system. A typical controller system regulates a control signal according to an error signal. The controller inputs can be the error, the rate of error and the integral error. The linguistic rules are the IF-THEN format as described in the following form;

Rule 1: IF the error is NB and the rate of error is ZE, THEN control is NB

Rule 2: IF the error is ZE and the rate of error is ZE, THEN control is ZE

Rule 3: IF the error is PB and the rate of error is ZE, THEN control is PB where positive big (PB), zero (ZE) and negative big (NB) are linguistic values defined by a fuzzy sets of the signal's error and rate of error. The IF-part is the *antecedent* part where rules are defined to describe the control system condition in terms of a combination of fuzzy propositions while the THEN-part is the *consequent* part which forms the desired control action of the output variable. Both *antecedent* and *consequent* can therefore be multiple term of multi-input-multi-output (MIMO) or single-input-single-output (SISO).

The Fuzzy Inference Mechanism is the main component of the fuzzy control that connects fuzzified inputs into output fuzzy sets in which the relevant control rules are evaluated at the current time and then decides what the controller output should be. There are two common methods of deductive inference for fuzzy system; (1) Mamdani systems and (2) Sugeno models.

In Mamdani system, two different ways can be used to find the rule conclusion namely 'Max-min' and 'Max-product' inference methods as illustrated in Figure 7-4.

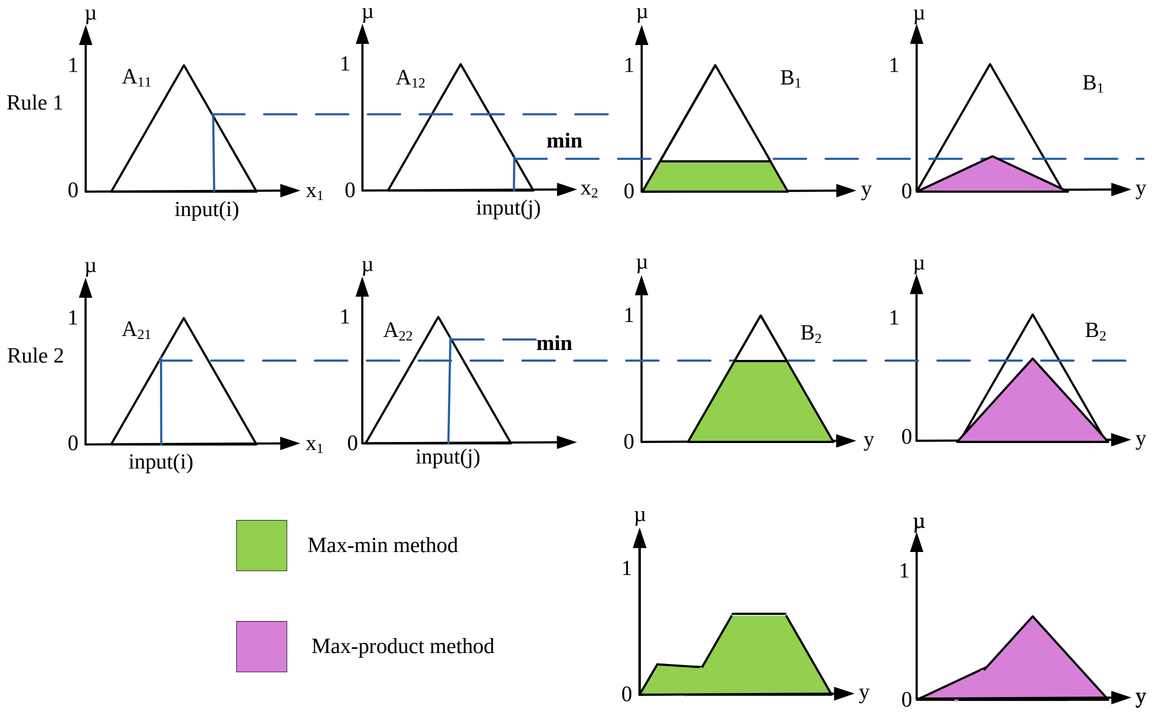


Figure 7-4 Example of Max-min and Max-product inference methods [Ross, 2004]

The 'Max-min' method uses the minimum function to combine the *antecedent* part of the IF-THEN rules which produces modified fuzzy sets for the outputs. These modified sets are then combined together using the maximum operator. The aggregated output of the *r* rules is given by [Ross, 2004];

$$\mu_{B^k} = \mathrm{max_k}[\mathrm{min}[\mu_{A_1^k}(i), \mu_{A_2^k}(j)]\,] \quad k = 1,2,.......r \qquad (7\text{-}1)$$

The aggregated output of the 'Max-product' (or correlation-product) method is slightly different to previous method as derived in following equation;

$$\mu_{B^k} = \mathrm{max_k}((i) \cdot \mu_{A_2^k}(j)) \quad k = 1,2,.......r \qquad (7\text{-}2)$$

The maximum operator is used to combine the product calculation of the *antecedent* part of the IF-THEN rules.

In contrast, the Sugeno method [Passino and Yurkovich, 1998; Ross, 2004] was proposed to systematically generate fuzzy rules from a given input-output data set, as follows;

IF *x* is A and *y* is B, THEN *z* is *z* = *f(x, y)*

where *z* = *f(x, y)* is a crisp function in the consequent part. The overall output of the system is obtained via a weighted average defuzzification, as shown in Figure 7-5. This approach avoids the time consuming methods of defuzzification necessary in the Mamdani system.

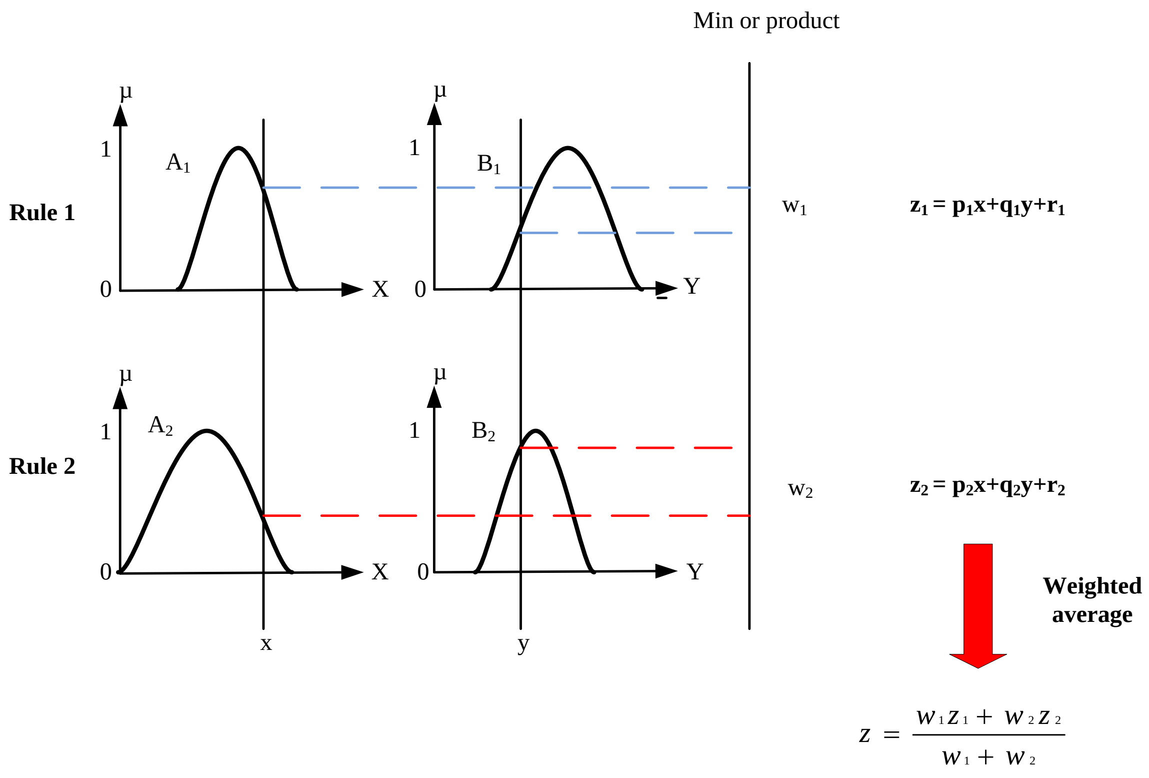


Figure 7-5 The Sugeno fuzzy model [Ross, 2004]

The defuzzification is a process used to evaluate each fuzzy output result and then convert the fuzzy quantity to the precise quantity. In the case of the Mamdani system, there are several techniques such as the maximum membership principle, centroid method, weighted average method and mean-max membership approach that were proposed to produce the crisp output. Among these methods, the weighted average method is the most frequently used in fuzzy applications since it is one of the most computationally efficient methods [Ross, 2004]. The weighted average method computes the weight of each output membership function by its respective membership value as given by the expression;

$$\mu^* = \frac{\sum \mu_c(x)x}{\sum \mu_c(x)} \tag{7-3}$$

As shown in Figure 7-6, the method is only restricted to symmetrical output membership functions.

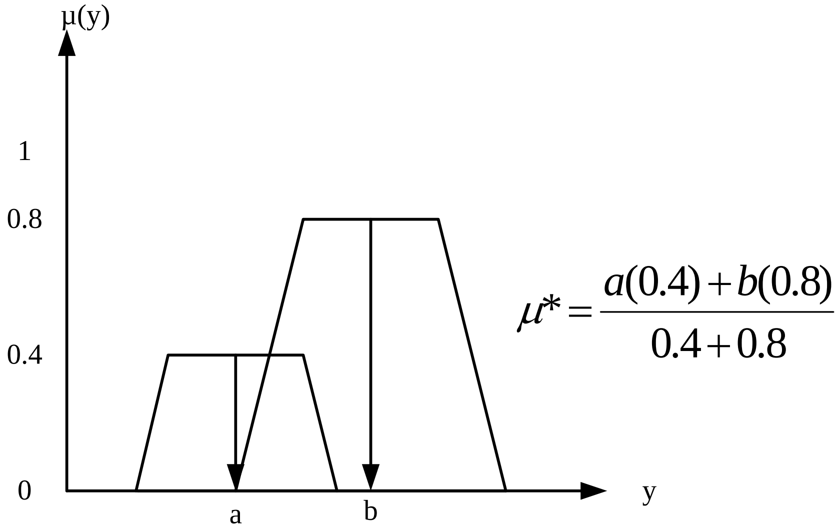


Figure 7-6 The weighted average method

Ross [2004] indicates that the Centre-of-Sums method is the faster defuzzification method and can be applied to any membership functions. This method is similar to the weighted average method except that the weight is the areas of the respective membership functions. Figure 7-7 provides an illustration of the centre of sums method and the defuzzified value that is given by the following equation;

$$\mu^* = \frac{\int \bar{x}\ \sum_{k=1}^{n} \mu_{c_k}(x)dx}{\int \sum_{k=1}^{n} \mu_{c_k}(x)dx} \tag{7-4}$$

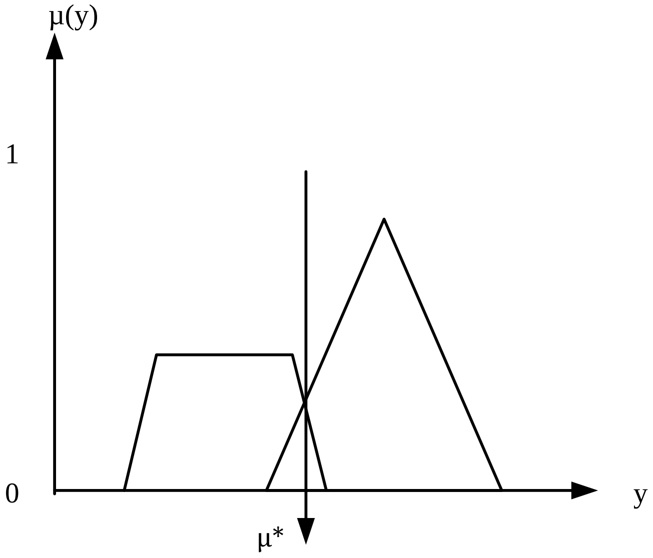


Figure 7-7 The Centre-of-Sums method

7.3 Artificial Neural Network

Artificial Neural Networks (ANNs) have gained a lot of interest in recent years as a powerful technique to solve many real world problems. Compared to conventional programming, the ANN has the unique capability of solving problems that do not have an algorithmic solution and are therefore found suitable for tackling problems that human are good at solving, such as pattern recognition.

Technically AANs are successfully applied in various areas such as classification and diagnostic (i.e fault diagnosis for electric motor), pattern recognition (i.e speech and handwriting character recognition), modeling (ability to represent both linear and non-linear relationship using historical data), forecasting and prediction (predict what will happen based on present information), estimation and control for system identification, adaptive control, parameter estimation and optimization [Driankov et al., 1996; Yen and Langari, 1999]. This is due to their high ability to learn from experience in order to improve their performance and to adapt themselves to changes in the environment in addition to their ability to deal with incomplete information or noisy data. They are also very effective particularly in situations where it is not possible to define the rules or steps that lead to the solution of a problem.

This section is not intended to present a full discourse on all neural network paradigms as the networks differ from one another in architecture and training algorithms. There are many fine texts and survey papers that present the broad scope of the field of neural networks [Lippmann, 1987; Wasserman, 1989; Müller et al., 1995; Yen and Langari, 1999; Gadoue, 2007]. The following section will discuss the general idea of ANN and how the technique can be implemented in this study.

7.3.1 Structure of Artificial Neural Networks

The structure of an ANN is similar to the biological neural cell which basically consists of a neuron or node with a summation and activation function as shown in Figure 7-8.

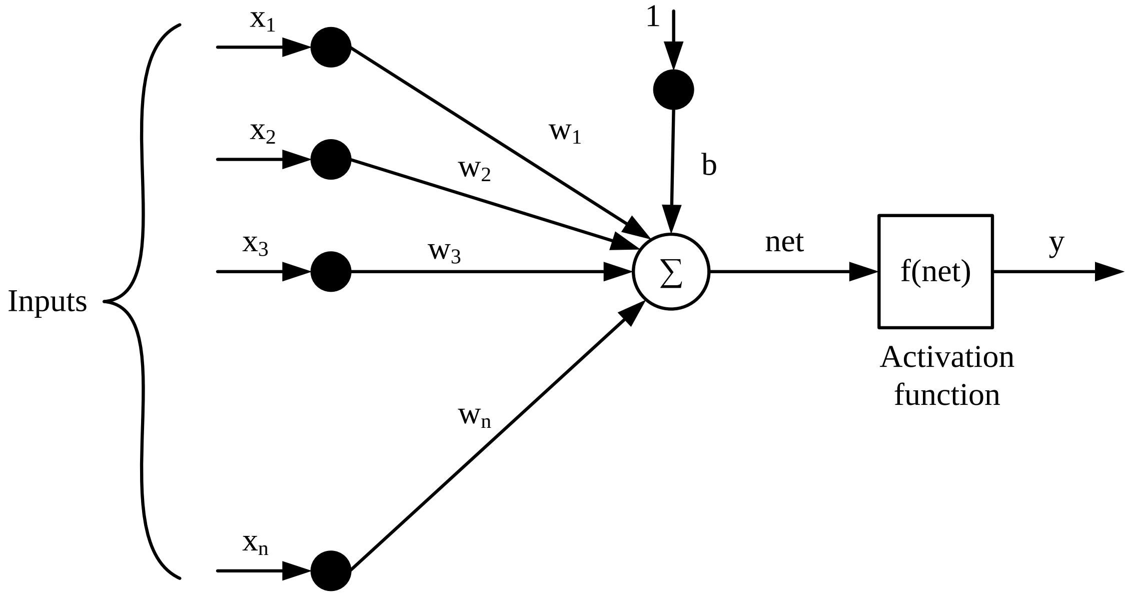


Figure 7-8 Structure of the artificial neuron [Gadoue, 2007]

where x_1, x_2, x_3 ……… x_n are the inputs to the neuron with assigned weights w_1, w_2, w_3 …w_n, and which model the synaptic neural connections in biological nets, and perform in such a way to increase and decrease the inputs signals to the neuron. A threshold term *b* is often added to the input. Generally, all inputs are multiplied by their corresponding weights and subsequently added together to form the input to the neuron called *net* whose mathematical expression can be written as;

$$net = \sum_{i=1}^{n} w_i x_i = w_1 x_1 + w_2 x_2 + w_3 x_3 + \ldots\ldots + w_n x_n \qquad (7\text{-}5)$$

The neuron behaves as an activation or mapping function *f(net)* to produce an output *y* which can be expressed as;

$$y = f(net) = f(\sum_{\mathrm{i=1}}^{\mathrm{n}} w_i x_i) \qquad (7\text{-}6)$$

where *f* is the neuron activation function or the neuron transfer function. Several functions such as linear, threshold, sigmoid and tansigmoid can be used to calculate the network output [Gadoue, 2007].

ANNs are widely used as nonlinear function approximators in control systems; the activation function is usually the multi-layer feedforward neural network which comprised of interconnected neurons organized in layers; input layer, hidden layer and output layer, as shown in Figure 7-9.

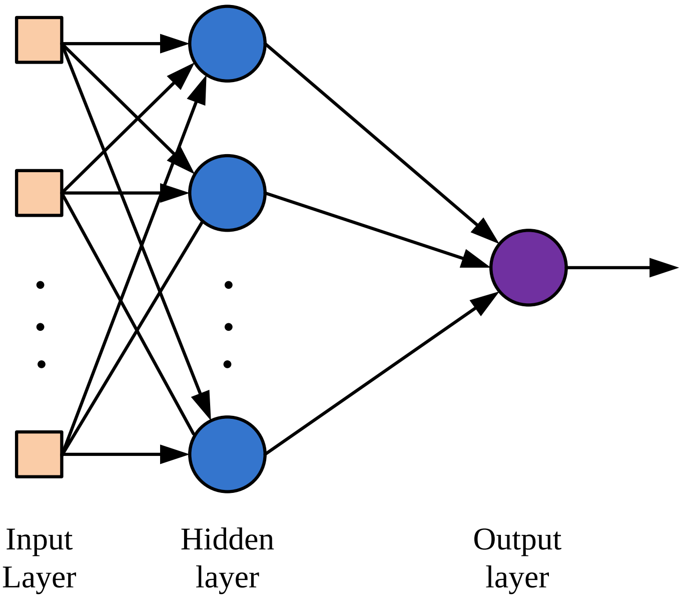


Figure 7-9 The architecture of three layer feedforward NN

An ANN can be classified into one of three main categories based on the adopted learning strategy; supervised, reinforcement or unsupervised learning. Only the supervised NN is discussed here as it is one of main components of Neuro-Fuzzy approach implementation. In principle, the supervised neural network (NN) can be imagined in a similar way to a teacher teaches a student. The NN acquires knowledge through a learning process based on a given input/target data training pattern associated with corresponding target or desired pattern. This type of NN is useful to learn a specific behaviour as well as the desired output for each given input target data. During the learning process, an error function, such as mean-squares error (MSE) is performed to minimize the average error between the neural network output and the target value. An optimal weight that represents the solution to the approximation problem can be optimized by the gradient descent algorithm called backpropagation.

7.3.2 How ANN is applied in fuzzy controller design?

Reznik [1997] discusses three main approaches to using ANNs in fuzzy controller design;

- Fuzzy systems where the ANN learns the shape of the surface of membership functions, the rules and output membership values;
- Fuzzy systems that are expressed in the form of ANN and are designed using a learning capability of the ANN;
- Fuzzy systems with ANN which are used to tune the parameters of the fuzzy controller as a design tool but not as a component of the final fuzzy system.

In this work, the second approach is applied in which ANN is a component of the whole Neuro-Fuzzy system. This will be discussed in detail is section 7.6.

7.4 Gain scheduling technique

A well tuned PI controller has parameters which are adapted to the dynamic properties of the control process in which the system can accomplish desired response and satisfactory stable performance. However, if the dynamic properties vary without re-tuning the controller, this condition leads to a degraded performance as a consequence of reduced stability in the control system. Previous studies have suggested that in many situations it is possible to observe how the dynamics of a process change with the operation conditions of the process and this dynamic change may be nonlinearities that are known [Åström and Wittenmark, 1995; Haugen, 2004]. The idea of *gain scheduling* in control system is to vary the gain parameters with variations of the dynamic process. Thus the performance of the system can be maintained independent to the operation conditions.

Prior to the application of artificial intelligent approaches to robot force control, a gain scheduling technique [Åström and Wittenmark, 1995; Jang and Sun, 1995] [Li and Gatland, 1995] [Haugen, 2004] [Zhen-Yu et al., 1993] was implemented to improve force control system performance. Optimal PI controller parameters might change depending on what “state” the system is in, and different gain parameters might be used once a certain set-point has been reached as the system responds differently once in the steady-state.

7.5 Design of Proportional-Integral (PI) Gain Scheduling

Gain scheduling (GS) is the process of modifying the gain parameters of a conventional P and I controller depending on the state of the system. These states can be defined by time duration or even by using system inputs. The method is very useful for systems that have predictable change in dynamics, so that predetermined gains can be calculated and applied. As discussed in Chapter 6, the conventional PI control algorithm (Equation 6.6) can be extended to the following equation;

$$\begin{aligned}\Delta x_r(t_k) &= F\{E(t_k), \Delta E(t_k)\} \\ &= F\{\,K_p \cdot \Delta e_f(t_k), K_i \cdot e_f(t_k)\}\end{aligned} \tag{7-7}$$

F signifies the function that acts on the rules given in the form of a gain scheduling look-up table. The concepts associated with the ‘fuzzy-like’ rule base are utilized to characterize the control rules and the data base. The strategy is subjectively defined and based on experience and engineering judgment from previous conventional method experiments. To reduce the subjectivity, the typical time step response of a control system is considered as shown in Figure 7-10. Type A and C are stable while B and D are unstable.

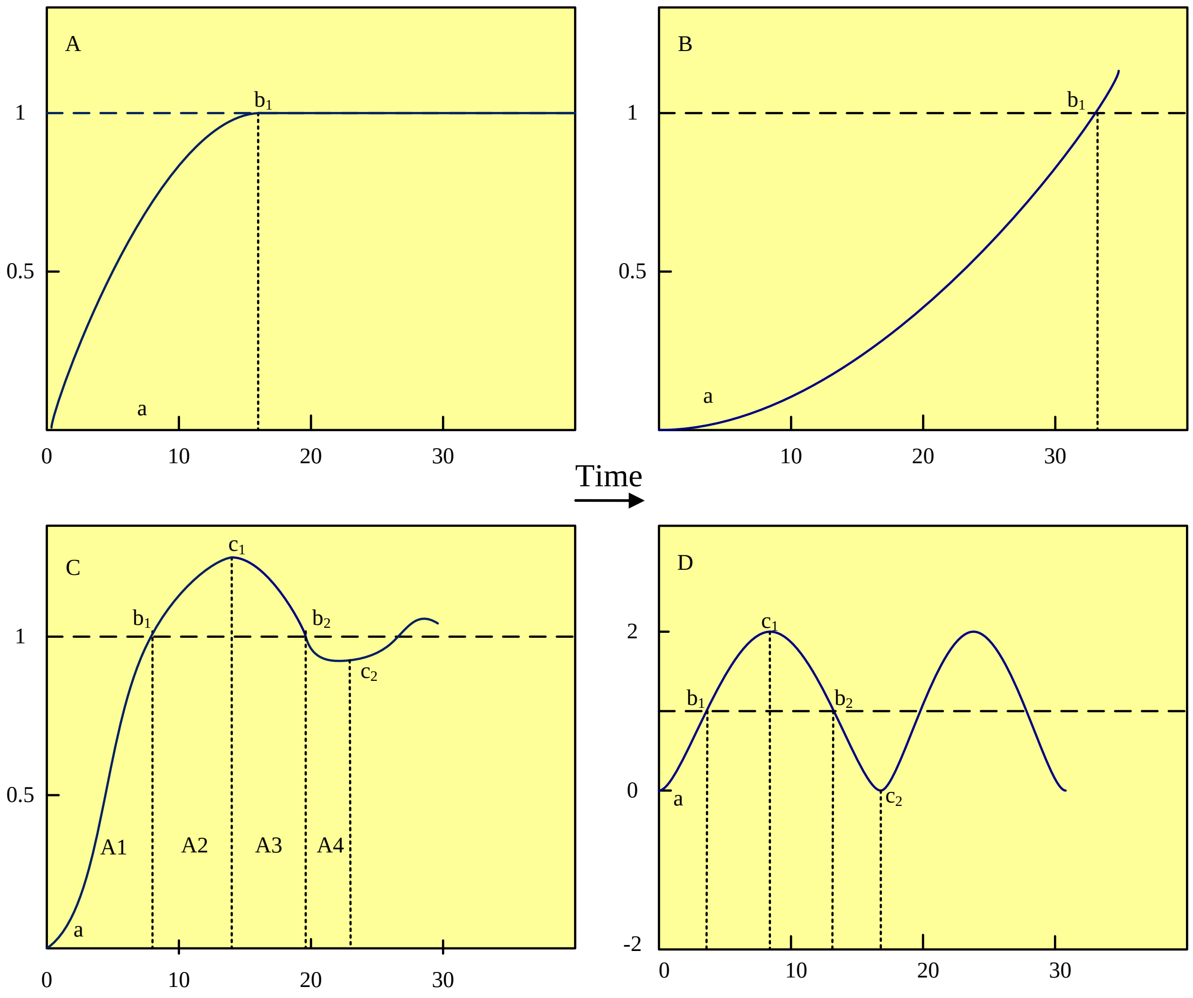


Figure 7-10 General type force step response

As can be seen from Figure 7-10, the type C response contains all the shape characteristics of the other three types. The type C response can be divided into four areas A1 to A4 and two sets point: cross over {b1,b2} and peak-valley {c1,c2} on the basis of the error *e* and rate of change of the error Δ*e* as listed below:

Response area

$A_1 : e > 0 \text{ and } \Delta e < 0$ $A_2 : e < 0 \text{ and } \Delta e < 0$

$A_3 : e < 0 \text{ and } \Delta e > 0$ $A_4 : e > 0 \text{ and } \Delta e > 0$

Cross over

$b_1 : e \geq 0 \text{ and } \Delta e < 0$ $b_2 : e \leq 0 \text{ and } \Delta e > 0$

Peak valley

$c_1 : e < 0$ and $\Delta e = 0$ $b_2 : e > 0$ and $\Delta e = 0$

Taken together, all possible system responses can be described by these four areas and two sets point. Type D response lies in all defined four areas, type A is in area A2 and type B occupies A1 area. As a result, the gain scheduling rule base can be described by following three metarules [Li and Gatland, 1995; Li and Gatland, 1996]:

1) If both e and Δe are zero, present control setting is maintained (Δx = 0).
2) If e varies around zero at satisfactory rate, present control setting is maintained (Δx = 0).
3) If e value is beyond acceptable rate, the control increment Δx is not zero and depends on the sign and magnitude of the e and Δe.

By analyzing the typical system step responses, the following detailed rules for the metarule (3) are obtained based on the system performance evaluation.

i. Rules for area A1 should increase the rise time when e is large and prevent the overshoot in area A2 when the response approaches the set point.
ii. The overshoot around the peak in area A2 should be decreased.
iii. Rules for area A3 mirror the rule for area A1.
iv. The overshoot around the valley in area A4 should be decreased.
v. The sign of the control increments also can be determined based on the previous listed rules as shown in following Table 7-1.

Table 7-1 Control increment signal sign

Rules No.	**Area/ set point**	**Error value**	**Control increment sign**
3.1	A1	a) $e > 0$ b) $e \approx 0$	positive zero or negative
3.2	A2	$0 < e < (c_1\text{-}b_1)$	negative
3.3	A3	a) $e > 0$ b) $e \approx 0$	negative zero or positive
3.4	A4	$(c_2\text{-}b_2) < e < 0$	positive
3.5	{b1, b2}	$e = 0$	same sign as Δe
3.6	{c1, c2}	$e = (c_1\text{-}b_1)$, $e = (c_2\text{-}b_2)$	same sign as e

7.5.1 PI gain scheduling force controller heuristic tuning strategy

Figure 7-11 shows the structure of the PI gain scheduling force control system. The gain scheduling variable, *GS* is a measured process condition based on previously derived rules which at every sample period represents the dynamic properties of the force control process. The most difficult problem with the method is the parameter tuning. The objective of tuning is to select the proper combination of PI controller parameters so that the resulting force controller response meets the desired control performance. The parameter values of the controller were determined by Ziegler-Nichols' closed loop method by means of an optimization routine, taking into account the tracking and rejection requirements in each defined area and set point.

It is assumed that the gains are in prescribed ranges $\{K_{p(min)}$ and $K_{p(max)}\}$ and $\{K_{i(min)}$ and $K_{i(max)}\}$ respectively. The appropriate ranges are determined experimentally as described in Chapter 6. The parameter values can be stored in a parameter table (the gain scheduler) as shown in Table 7-2. From the table, proper PI parameters are given as functions of the gain scheduling variable, *GS*.

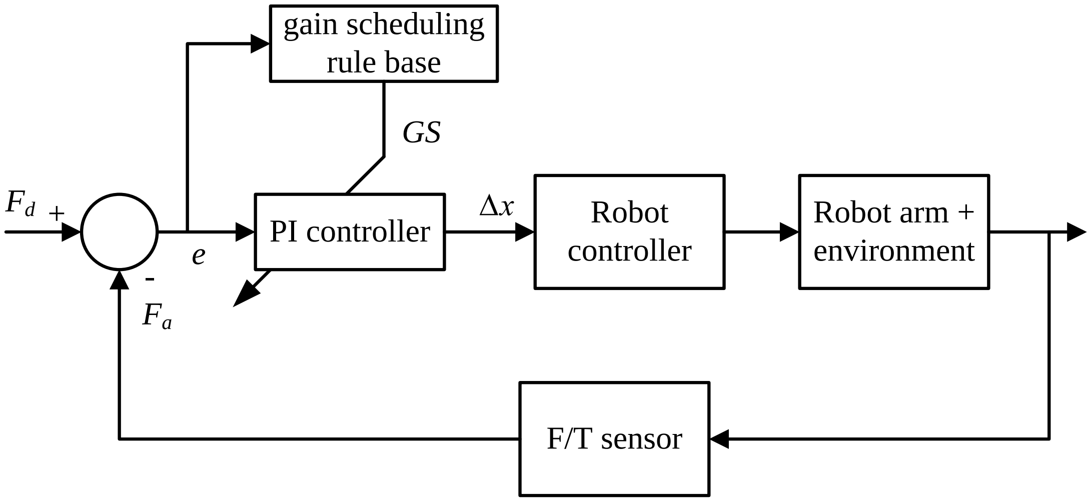


Figure 7-11 PI Gain Scheduling force controller

Table 7-2 The Gain Scheduler

Gain scheduling variable	Gain scheduling rule base	Proportional gain	Integral gain
$\boldsymbol{GS_1}$	1	K_{p1}	K_{i1}
$\boldsymbol{GS_2}$	2	K_{p2}	K_{i2}
$\boldsymbol{GS_3}$	3.1 (a)	K_{p3}	K_{i3}
$\boldsymbol{GS_4}$	3.1 (b)	K_{p4}	K_{i4}
$\boldsymbol{GS_5}$	3.2	K_{p5}	K_{i5}
$\boldsymbol{GS_6}$	3.3 (a)	K_{p6}	K_{i6}
$\boldsymbol{GS_7}$	3.3 (b)	K_{p7}	K_{i7}
$\boldsymbol{GS_8}$	3.4	K_{p8}	K_{i8}
$\boldsymbol{GS_9}$	3.5	K_{p9}	K_{i9}
$\boldsymbol{GS_{10}}$	3.6	K_{p10}	K_{i10}

An interval is defined for each of the *GS* variable in the parameter table. The controller is kept constant as long as the *GS* is within the defined range. The range is determined based on the current error *e* and derivative error Δ*e* using a set of rules of the form

If *e(k)* is *Ai* and Δ*e(k)* is *Bi*, then output is *GSi*

Here, A_i and B_i are derived by the operator experience and knowledge based on previous step response experiments. For example, at initial state of the response (i.e. in Figure 7-10(A)), a large control signal is required to achieve a fast rise time. Hence, the PI controller should have a large proportional gain. Around point b_1, a small increment signal is expected in order to avoid a large overshoot, thus, the PI controller should have small proportional and integral gains which can smooth the response around the set point. When the *GS* variable changes from one interval to another interval the controller parameters are changed immediately, as shown in Figure 7-12 for proportional gain, K_p.

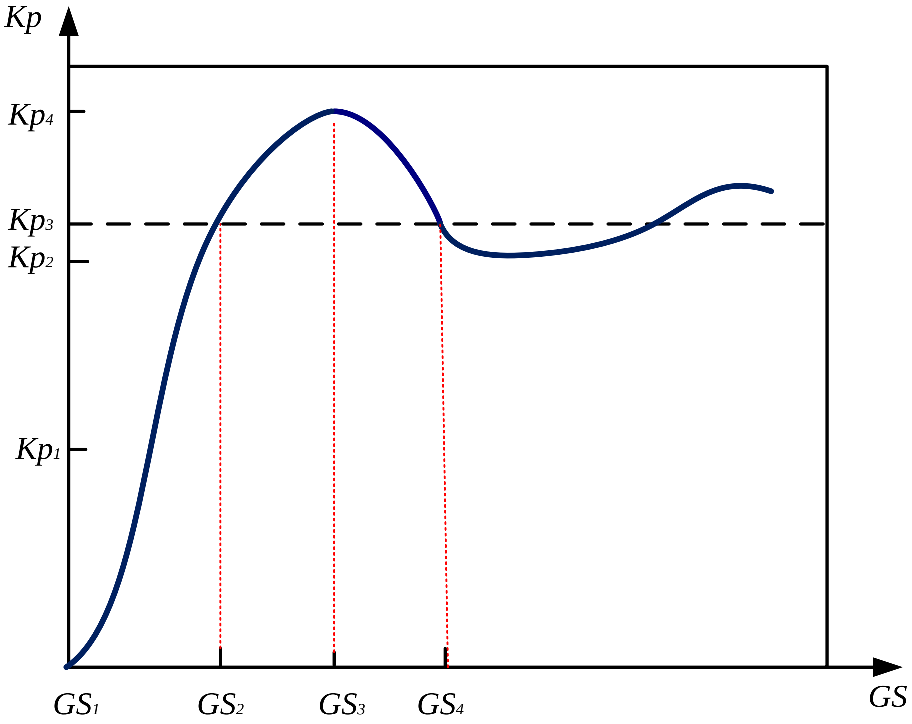


Figure 7-12 Gain scheduling variables changes

The proportional and integral gains can be expressed mathematically as follows;

$$K_p(t) = K_{p(max)} - (K_{p(\max)} - K_{p(\min)})exp^{-(a|e(t)|)} \tag{7-8}$$

where a is constant and $K_{p(max)}$ and $K_{p(min)}$ are the maximum and minimum values of the proportional gain K_p. From equation (7-8), it can be seen that when $e(t)$ is large the exponential term approach zero and therefore K_p is $K_{p(max)}$. Similarly when error $e(t)$ is small the exponential term approaches 1 and hence the K_p is $K_{p(min)}$.

The K_i gain varies in the range $K_{i(min)} < K_i < K_{i(max)}$. For instance, when the system is in the steady-state error period in which the error $e(t)$ is large, K_i is set to minimum as the control is fully depended on the $K_{p(max)}$ to achieve desired control value, alternatively the K_i is increased to its maximum value to overcome steady-state error during steady-state period to reduce oscillation and overshoot. Based on the described Table 7-2 and equation (7-8), the PI parameters are varied online as a function of the force error e and rate of error Δe. This approach should improve the transient response and steady-state performance of the PI controller.

7.5.2 Performance evaluation

The following parameters, derived from Ogata [2009] and Dorf and Bishop [2008], are used to evaluate the robustness and stability of robot force control performance, namely:

Delay time, t_d - indicates the time required for the response to reach half of the final value at the very first time.

Rise time, t_r – indicates the time required for the response to rise from 10% to 90%, 5% to 95% or 0% to 100% of its final value. For underdamped second-order systems the rise time of 0% to 100% is normally used.

Peak time, t_p – specifies the time required for the response to reach the first peak of the overshoot.

Maximum (percent) overshoot, M_p – calculates the maximum peak value of the response curve measured form unity. The value also indicates the relative stability of the system. M_p is defined by

$$M_p = \frac{M_{pt} - f_v}{f_v} \times 100\% \tag{7-9}$$

where M_{pt} is the peak value of the time response and f_v is the final value of the response.

Settling time, t_s – defines the time required for the system to settle with a certain percentage (usually 2% to 5%). This parameter criterion is useful to determine the closeness of the response to the desired response. Figure graphically represents the performance specifications.

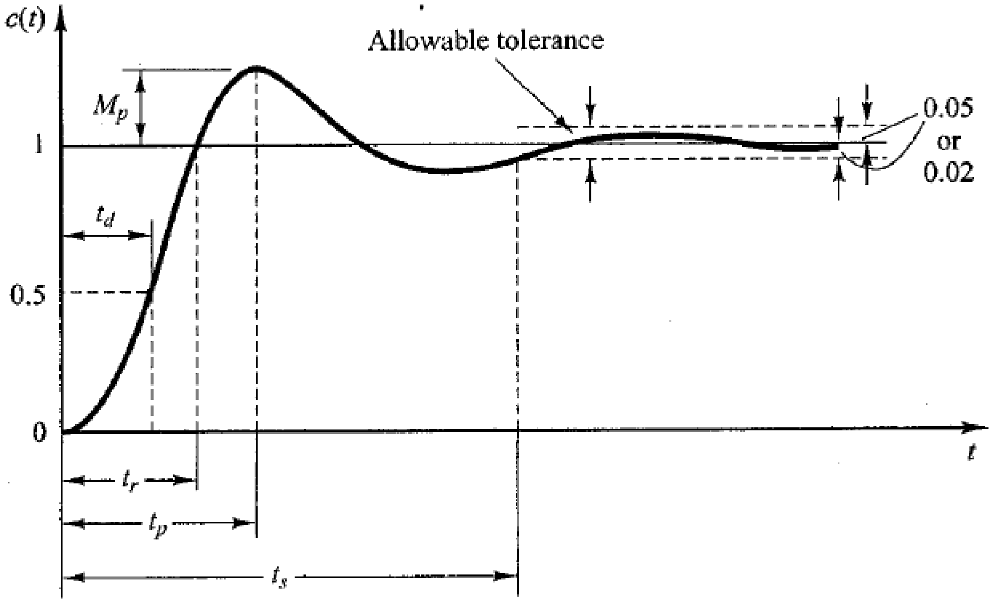


Figure 7-13 Unit-step response of control system [Dorf and Bishop, 2008; Ogata, 2009]

In addition, a performance index (quantitative measure) is chosen to emphasis the system performance. The IAE (Integral of Absolute Error) and ITAE (Integral of Time Absolute Error) were selected in which the IAE accounts mainly for error at the beginning of the response while the ITAE evaluation not only give quantitative measure of the performance at beginning but also emphasizes the steady state error. The lower the measure of these criteria means better force tracking. Equation (7-11) provides the IAE and ITAE equations for the force control;

$$IAE_F = \int |e| dt \tag{7-10}$$

$$ITAE_F = \int t\,|e| dt \tag{7-11}$$

7.5.3 Test of PI gain scheduling technique description

A similar step input test response as discussed in Chapter 6 was repeated and recorded in Figure 7-14. All control parameters, including the environment stiffness and robot arm configuration are identical to the test response shown in Figure 5-6. Thus a direct comparison between conventional PI and PI gain scheduling technique can be drawn, illustrating the viability of the proposed technique. The results obtained from employing both methods will evaluate using the performance measures in section 7.5.2.

7.5.3.1 *Test results and discussion*

Figure 7-14 illustrates the comparison step response results of conventional PI and PI gain scheduling methods. Clearly, the response to the PIGS implementation particularly at initial portion of the transient marked ‘A’ shows a significant improvement of force response. This is confirmed by performance analysis of Table 7-3 that the delay, rise time and settling time had reduced to 85%, 83% and 89% respectively. This improvement is a result of a correct tuning of the gain parameters for each defining dynamic response ‘state’.

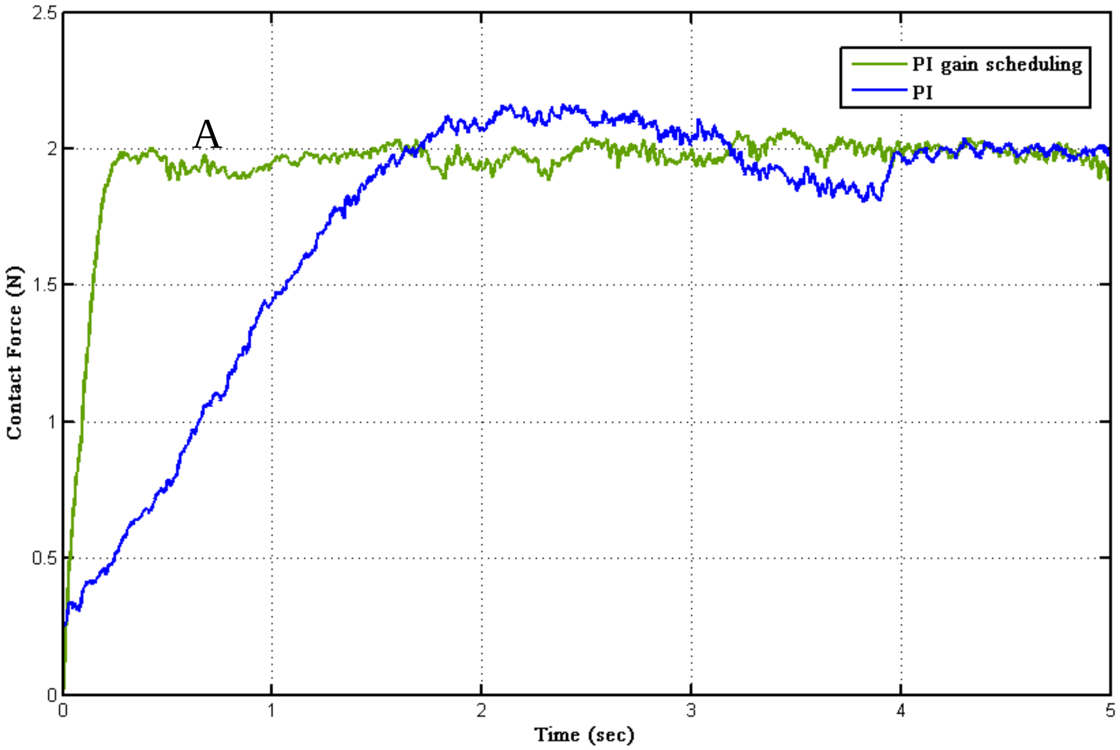


Figure 7-14 Comparison of step response of PI gain scheduling and PI force controller for environment of 10N/mm

Table 7-3 PI and PI gain scheduling force controller performance evaluation

Controller	t_d (s)	t_r (s)	t_p (s)	M_p	t_s (s)	IAE (N)	ITAE (N.s)
PI	0.65	1.67	2.34	10%	4.00	402.25	1609.00
PIGS	0.10	0.28	-	-	0.41	141.60	566.40

As can be seen clearly form the graph, a better steady-state performance is achieved as overshoot and oscillation behaviour is progressively diminished. It can be observed that the maximum overshoot is almost zero or not noticeable during the test. The low performance index values of IAE and ITAE provide strong evidence that a stable and faster force control can be achieved with the PIGS technique (the system is improved by 30 % compared to the conventional PI method).

Satisfactory results were obtained using the PI gain scheduling implementation. Human knowledge and experience in control system design is exploited in the tuning PI parameters as described in section 7.5. Although a rule of thumb for choosing the ranges for K_p and K_i was experimentally and heuristically obtained as described in section 7.5.1, it was still possible considered to make further performance improvements by fine tuning the ranges as well as by modifying the tuning rules in section 7.5.1 using combination of fuzzy logic and neural network approaches. Another possible complication using gain scheduling is that the range transition might lead to instabilities if the PI controller is not designed to make smooth transitions. Furthermore, the gain scheduling method also requires tuning at each operating range which is very cumbersome for different environment stiffness. The following sections will discuss further issue concerning of the intelligent control system in the force controller.

7.6 Neuro-Fuzzy Modelling

As previously discussed, the basic structure of the Fuzzy Inference System (FIS) is a model that maps input characteristics to input Membership functions (MFs), input MFs to rules, rules to a set of output characteristics, output characteristics to output MFs and the output MFs to a single-valued output or a decision associated with the output. Normally only fixed MFs that were chosen arbitrarily were considered. Generally, FIS is only applied to a modeling system whose rule structure is essentially predetermined by the user's interpretation of the characteristics of the variables in the model. This has led to the concept of applying a neural network to an efficient learning algorithm which can be helpful in tuning the fuzzy system.

An adaptive network fuzzy inference system (ANFIS) works in a different way from other FIS in which the structure is similar to neural network. The network can be interpreted as the input/output map in which inputs are mapped through inputs MFs and associated parameters and then subsequently through output MFs and associated parameters to the outputs.

The parameters associated with the membership functions changes through the learning process. The computation of these parameters (or their adjustment) is facilitated by a gradient vector. This gradient vector provides a measure of how well the fuzzy inference system is modeling the input/output data for a given set of parameters. When the gradient vector is obtained, several optimization routines can be applied in order to adjust the parameters to reduce some measure of the system error. This measure is usually defined by a least square estimation difference between actual and desired outputs. ANFIS can use either back propagation or a combination of least squares estimation and backpropagation for membership function parameter estimation. These learning algorithms have been applied and widely discussed; see the references by [Jang, 1993; Jang and Sun, 1995; Yen and Langari, 1999; Gadoue, 2007].

7.6.1 ANFIS: Adaptive Neuro-Fuzzy Inference System

Jang and Sun [1993] introduce an adaptive network-based fuzzy inference system in which the Takagi-Sugeno fuzzy model is represented as a multilayer feedforward neural network with supervised learning capability. The adaptive network is a structure that consists of nodes and directional links through which the nodes are connected, as shown in Figure 7-15.

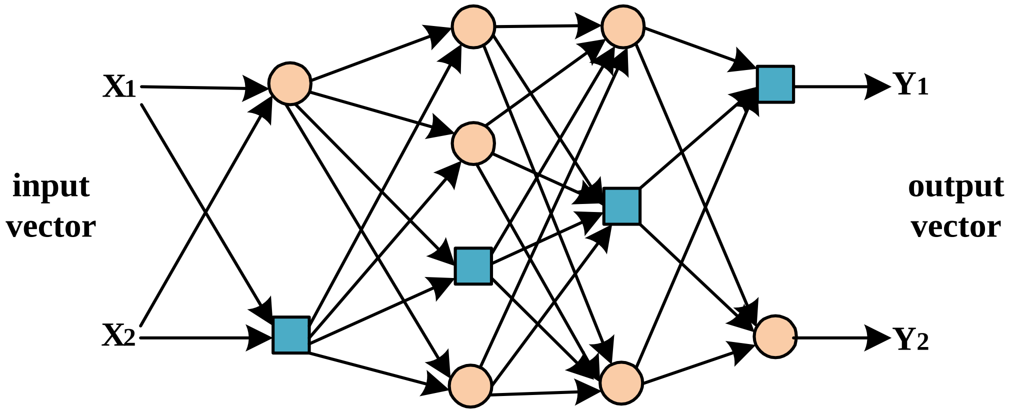


Figure 7-15 Neuro-Fuzzy Adaptive Network

Each node performs a particular function (node function) on all incoming signals as well as a set of parameters related to this node. The flow direction of the signals between nodes is indicated by the links in the network from left to right. Both circle (fixed node) and square (adaptive node) nodes are used to represent different adaptive capabilities. Note that only an adaptive node has parameters that are updated according to given training data and a learning procedure as described in following section 7.6.2.

7.6.2 ANFIS Architecture

In order to realize a desired force response in an unknown stiffness environment, an adaptive network-based fuzzy inference system [Jang, 1993; Jang and Sun, 1995; Jang et al., 1997] is adopted to have the adaption and estimation ability. The controller is designed based on human expert knowledge using linguistic rules which are adjusted online to adapt to the unknown environment and compensate for any external disturbances.

Figure 7-16 illustrates the Neuro-Fuzzy force controller architecture that was implemented on the TMS robotic system. The Neuro-Fuzzy force controller has two inputs - the force error signal *e* and the rate of force error signal Δe, and one output - the desired incremental robot displacement Δx. Seven fuzzy linguistic variables (NB, NM, NS, ZE, PS, PM, PB) were adopted from the concept of PI gain scheduling controller.

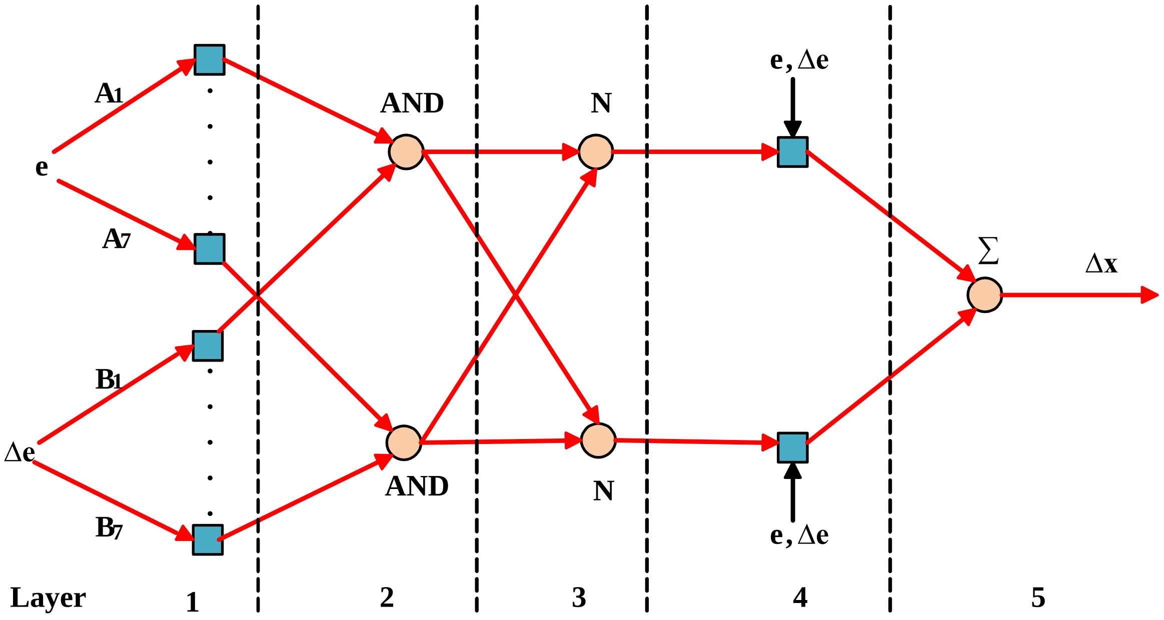


Figure 7-16 Neuro-Fuzzy force controller architecture

The fuzzy inference system is the Takagi and Sugeno's type (TS) in which the output is derived as a first order linear equation. Without loss of generality and assuming 1st order TS type with the following two rules

If e is $\mathbf{A_1}$ and Δe is $\mathbf{B_1}$,

then the output is $\boldsymbol{p_{11}*e+q_{12}*\Delta e+r_{10}}$

If e is $\mathbf{A_2}$ and Δe is $\mathbf{B_2}$,

then the output is $\boldsymbol{p_{21}*e+q_{22}*\Delta e+r_{20}}$

As shown in Figure 7-16, the 1st layer (input layer) is an adaptive node that contains membership functions (MFs) with the node output is a matching degree of an input to the corresponding MFs in the fuzzy set.

$$O_{1i} = \mu_{Ai}(e), \quad i = 1, 2.....7$$

$$O_{1i} = \mu_{Bi}(\Delta e), i = 8, 9.....14$$

where the O_{1i} is the MFs grade of a fuzzy set A_i (force error) and B_i (rate of force error). The $\mu_{Ai}(e)$ and $\mu_{Bi}(\Delta e)$ are defined by Gaussian-shaped MFs as

$$\mu_i(x) = \mathrm{e}^{\frac{-(\mathrm{x}-\mathrm{c_i})^2}{2\sigma_\mathrm{i}{}^2}}$$

where $\{c_i, \sigma_i\}$ are called premise parameters. The Gaussian-shaped functions vary as the values of these parameters change, thus exhibiting various forms of MFs on linguistic labels A_i and B_i.

Every node in the 2nd layer (Fuzzification layer) is a fixed AND node whose output is the product of the incoming signals and represents the firing strength of a rule.

$$O_{2i} = w_i = \mu_{Ai}(e) * \mu_{Bi}(\Delta e) \qquad i = 1,2 \ldots \ldots 49$$

The 3rd layer (Fuzzy-Rule layer) is called the normalized node which calculates the ratio of the ith rules's firing strength to the sum of all rules's firing strength:

$$O_{3i} = \overline{w_\iota} = \frac{w_i}{w_1 + w_2} \qquad i = 1,2 \ldots \ldots \ldots 49$$

Every node in the 4th layer or Defuzzification layer is an adaptive node, calculated by

$$O_{4i} = \overline{w_\iota} f_i = \overline{w_\iota}(p_i e + q_i \Delta e + r_i)$$

where $\overline{w_\iota}$ is the output of layer 3 and $\{p_i,q_i,r_i\}$ is referred to as consequent parameters.

Finally, the single node in the 5th layer (output layer) is a fixed node which computes summation of the overall 4th layer outputs.

$$O_{5i} = \sum_i \overline{w_\iota} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}$$

There are two main sets of parameters for the Neuro-Fuzzy system which are MFs on the 1st layer and first order linear function in 4th layer. Several techniques including reinforcement learning, stochastic learning and error-correction learning can be used to train these sets of parameters. A hybrid learning algorithm consisting of combination of the least-square method (LSE) and the backpropagation gradient descent method is employed on the basis that the method converges much faster since it reduces the dimension of the search space of the original back-propagation method.

7.7 Design of Neuro-Fuzzy Force Controller

This section will describe the heuristic process of tuning Neuro-Fuzzy force controller. After adjustment and validation of all parameters, Neuro-Fuzzy controller was applied to the robotic system. Comparisons between three methods (PI, PI gain scheduling and Neuro-Fuzzy) were conducted using step response tests.

7.7.1 Tuning Neuro-Fuzzy Force Controller

As discussed in Chapter 6, the environmental parameters are known exactly and the gain can be determined in such a way that the desired contact force can be maintained precisely. In this study, it is assumed that the exact location and the stiffness of the environment are unknown but a limit of both parameters is considered wide enough. An adaptive Neuro-Fuzzy control approach is utilized in order to tune the force controller. The approach is designed to ensure that the robot can maintain stable force control under unknown environment conditions using expert knowledge. The Neuro-Fuzzy force controller has two inputs: the force error signal and the derivative of error signal and one output: the desired incremental robot displacement. The adaptive Neuro-Fuzzy force controller was developed as follows: The Neuro-Fuzzy force controller input variables are first normalized into seven linguistic labels which are negative big (**NB**), negative medium (**NM**), negative small (**NS**), Zero (**ZE**), positive small (**PS**), positive medium (**PM**) and positive big (**PB**) which leads to 49 rules as shown in Table 7-4. The fuzzy inference system is of Takagi and Sugeno type, thus the output can be derived as a first order linear equation.

Table 7-4 Inputs MF variables, output parameters and rule-base of Neuro-Fuzzy force controller

e Δe	MFs parameters (width, centre)	**NB**	**NM**	**NS**	**ZE**	**PS**	**PM**	**PB**
		(σ_8, c_8)	(σ_9, c_9)	(σ_{10}, c_{11})	(σ_{12}, c_{12})	(σ_{13}, c_{13})	(σ_{14}, c14)	(σ_{15}, c_{15})
NB	(σ_1, c_1)	(p_1, q_1, r_1)	(p_2, q_2, r_2)	(p_3, q_3, r_3)	(p_4, q_4, r_4)	(p_5, q_5, r_5)	(p_6, q_6, r_6)	(p_7, q_7, r_7)
NM	(σ_2, c_2)	(p_8, q_8, r_8)	(p_9, q_9, r_9)	(p_{10}, q_{10}, r_{10})	(p_{11}, q_{11}, r_{11})	(p_{12}, q_{12}, r_{12})	(p_{13}, q_{13}, r_{13})	(p_{14}, q_{14}, r_{14})
NS	(σ_3, c_3)	(p_{15}, q_{15}, r_{15})	(p_{16}, q_{16}, r_{16})	(p_{17}, q_{17}, r_{17})	(p_{18},q_{18},r_{18})	(p_{19}, q_{19}, r_{19})	(p_{20}, q_{20}, r_{20})	(p_{21}, q_{21}, r_{21})
ZE	(σ_4, c_4)	(p_{22}, q_{22}, r_{22})	(p_{23}, q_{23}, r_{23})	(p_{24}, q_{24}, r_{24})	(p_{25}, q_{25}, r_{25})	(p_{26}, q_{26}, r_{26})	(p_{27}, q_{27}, r_{27})	(p_{28}, q_{28}, r_{28})
PS	(σ_5, c_5)	(p_{29}, q_{29}, r_{29})	(p_{30}, q_{30}, r_{30})	(p_{31}, q_{31}, r_{31})	(p_{32}, q_{32}, r_{32})	(p_{33}, q_{33}, r_{33})	(p_{34}, q_{34}, r_{34})	(p_{35}, q_{35}, r_{35})
PM	(σ_6, c_6)	(p_{36}, q_{36}, r_{36})	(p_{37}, q_{37}, r_{37})	(p_{38}, q_{38}, r_{38})	(p_{39}, q_{39}, r_{39})	(p_{40}, q_{40}, r_{40})	(p_{41}, q_{41}, r_{41})	(p_{42}, q_{42}, r_{42})
PB	(σ_7, c_7)	(p_{43}, q_{43}, r_{43})	(p_{44}, q_{44}, r_{44})	(p_{45}, q_{45}, r_{45})	(p_{46}, q_{46}, r_{46})	(p_{47}, q_{47}, r_{47})	(p_{48}, q_{48}, r_{48})	(p_{49}, q_{49}, r_{49})

If e is **NS** and Δe is **PB**, then the output is ***pi*e+qi**Δ*e+ri***

Firstly, a set of Gaussian membership functions as described in section 7.6.2 were chosen to represent the desired controller inputs. Then, a rule-base table of PI gain scheduling controller implementation (section 7.5) was adopted to set up initial FIS parameters. Step force response tests were carried out to acquire real data for training the artificial neural network. A hybrid learning algorithm consisting of a combination of the least-squares method (LSE) and the backpropagation gradient descent method was used for training the FIS membership function parameters. The training routine was carried out offline using the Adaptive Neuro-Fuzzy Inference System (ANFIS) MATLAB® toolbox [MathWorks, 2011] using the training data from the tests. After adjustment and validation of all the parameters the Neuro-Fuzzy controller was applied to the robot force controller for further experimental tests. Figure 7-17 illustrates a flow chart of a heuristic method to design the Neuro-Fuzzy force controller.

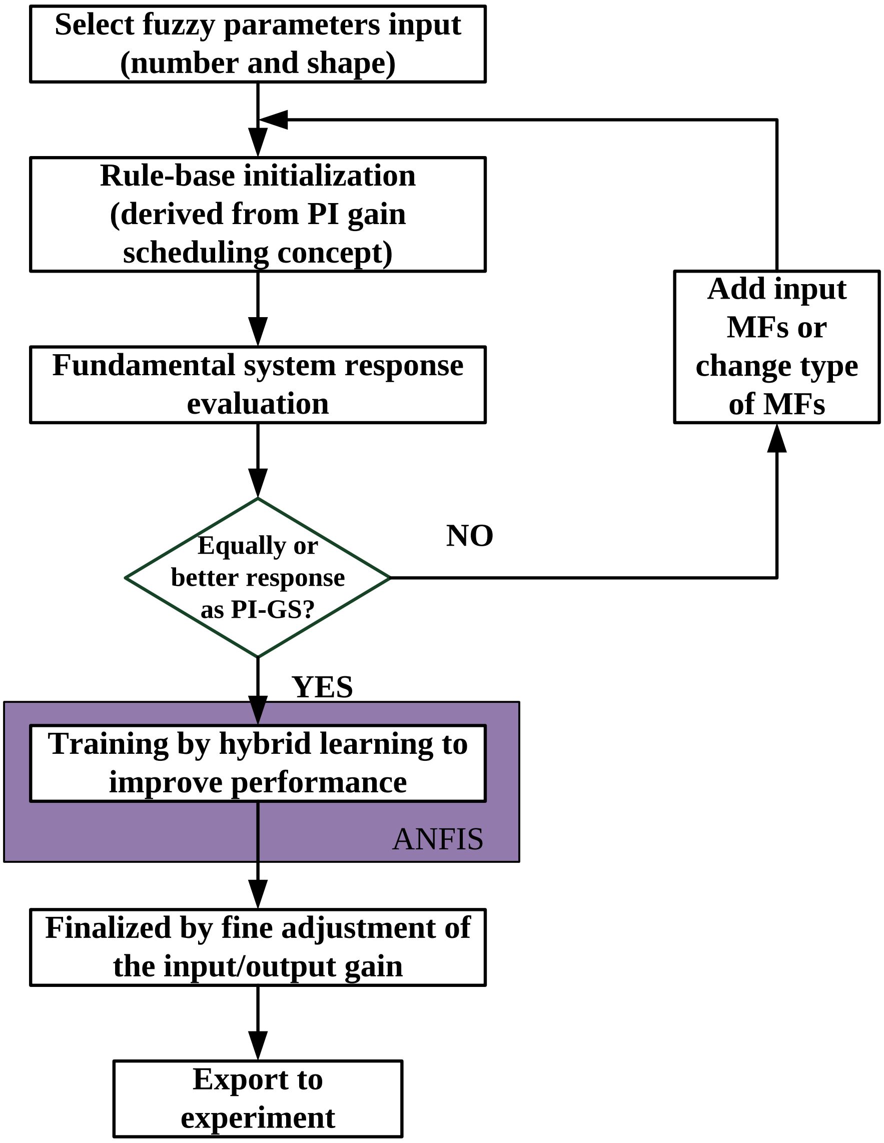

Select fuzzy parameters input (number and shape)
Rule-base initialization (derived from PI gain scheduling concept)
Fundamental system response evaluation
Equally or better response as PI-GS?
NO
Add input MFs or change type of MFs
YES
Training by hybrid learning to improve performance
ANFIS
Finalized by fine adjustment of the input/output gain
Export to experiment

Figure 7-17 Neuro-Fuzzy force controller tuning procedure

Figure 7-18 provides a mesh plot of relationship between controller inputs (force error and derivative of force error) and controller output (the desired incremental robot position). This control surface plot shows that the nonlinear neuro-fuzzy controller as an effective gain scheduler technique. A steep surface implies a higher gain near the centre of force error and derivative of force error, which helps to reduce the error more quickly when the error is large. When the response is nearly force setpoint, the controller output has decreased demonstrated by a flat surface to avoid any possible overshoot or undesired behaviour.

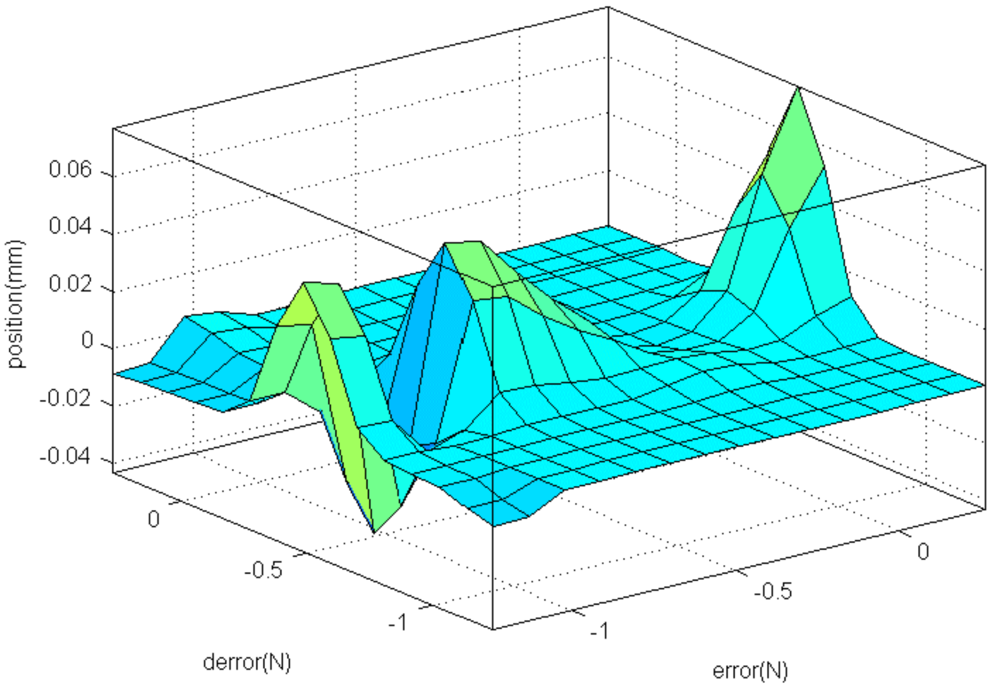


Figure 7-18 Control surface of Neuro-Fuzzy Force Controller

7.7.1.1 *Neuro-Fuzzy Force Controller test results and discussion*

Step response test in section 7.5.3 was repeated and the comparison results are provided in Figure 7-19. Similar improved response characteristics to that obtained in Figure 7-14 were found using both PI gain scheduling and Neuro-Fuzzy (NF) methods. To compare the performance of the three methods, the performance evaluation parameters discussed in section 7.5.2 was used (refer to the Table 7-5). Figure 7-19(c) shows that the NF performance is closely analogous to the PIGS (Figure 7-19(b)) and the response is evidently improved. This is expected as the Neuro-Fuzzy force controller is trained using the test data of PIGS implementation.

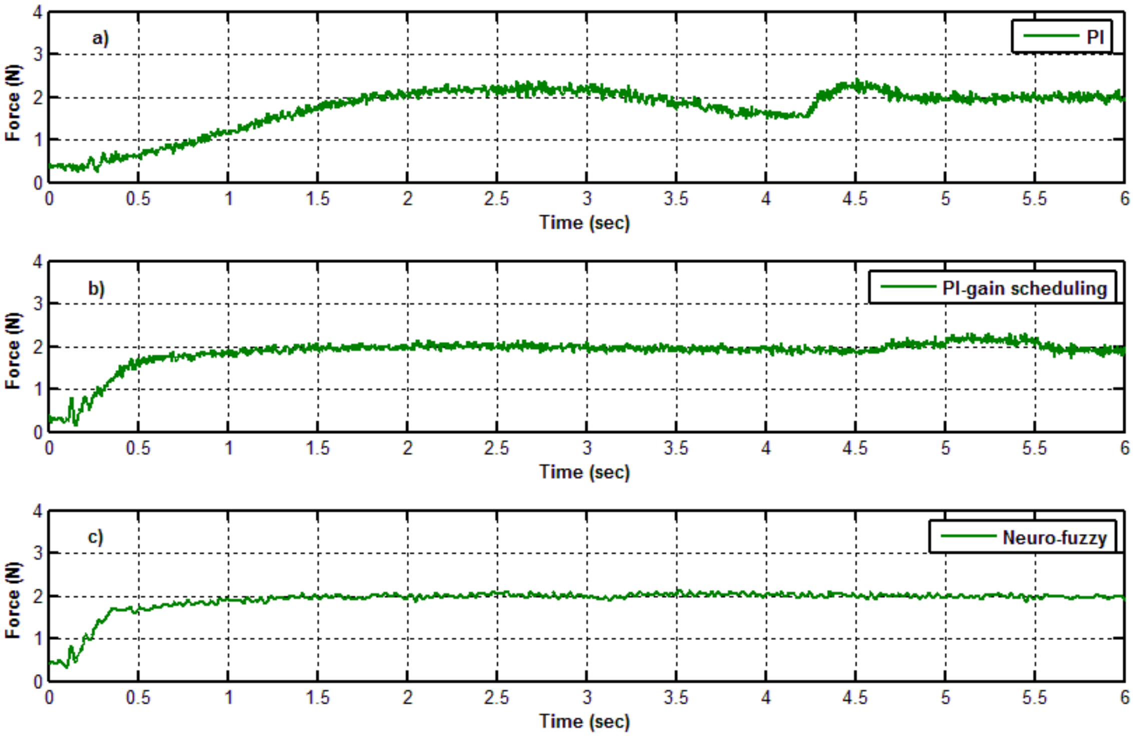


Figure 7-19 Comparison of step response of (a) PI, (b) PI gain scheduling and (c) Neuro-Fuzzy force controller, k_e = 10N/mm

Table 7-5 PI, PIGS and NF force controller performance evaluation of k_e = 10N/mm

Controller	t_d (s)	t_r (s)	t_s (s)	IAE (N)	ITAE (N.s)
PI	0.83	1.832	4.74	222.39	2039.7
PIGS	0.284	1.076	1.088	156.99	949.28
NF	0.212	0.956	1.008	111.77	774.03

In agreement with the improved performance of the PIGS, the Neuro-Fuzzy technique were shown experimentally to reduce delay time, rise time and settling time of transient response approximately 25%, 11% and 7% respectively compared to the PIGS technique as can be seen in Table 7-5. The other important observation is the reduced of uncontrollable random signal noise displayed in Figure 7-19(a) and virtually eradicated in Figure 7-19(c) resulting a low value of force error as indicated by the IAE and ITAE index of 111.77N and 774.03N.s respectively. These values shows improvement of almost 50% compare to conventional force control law in which display an important benefit of intelligent approach in helping to overcome the force control problem.

To confirm the significance of the Neuro-Fuzzy approach, a similar test to that shown in Figure 7-19 was repeated and recorded. Identical PIGS and NF controllers were used in both tests. The tests of Figure 7-20 differed from Figure 7-19 in that the environment stiffness is much stiffer which approximately 50N/mm. Figure 7-20(a) shows the same information as presented in Figure 7-19(a) and indicates poor behaviour of the conventional force controller.

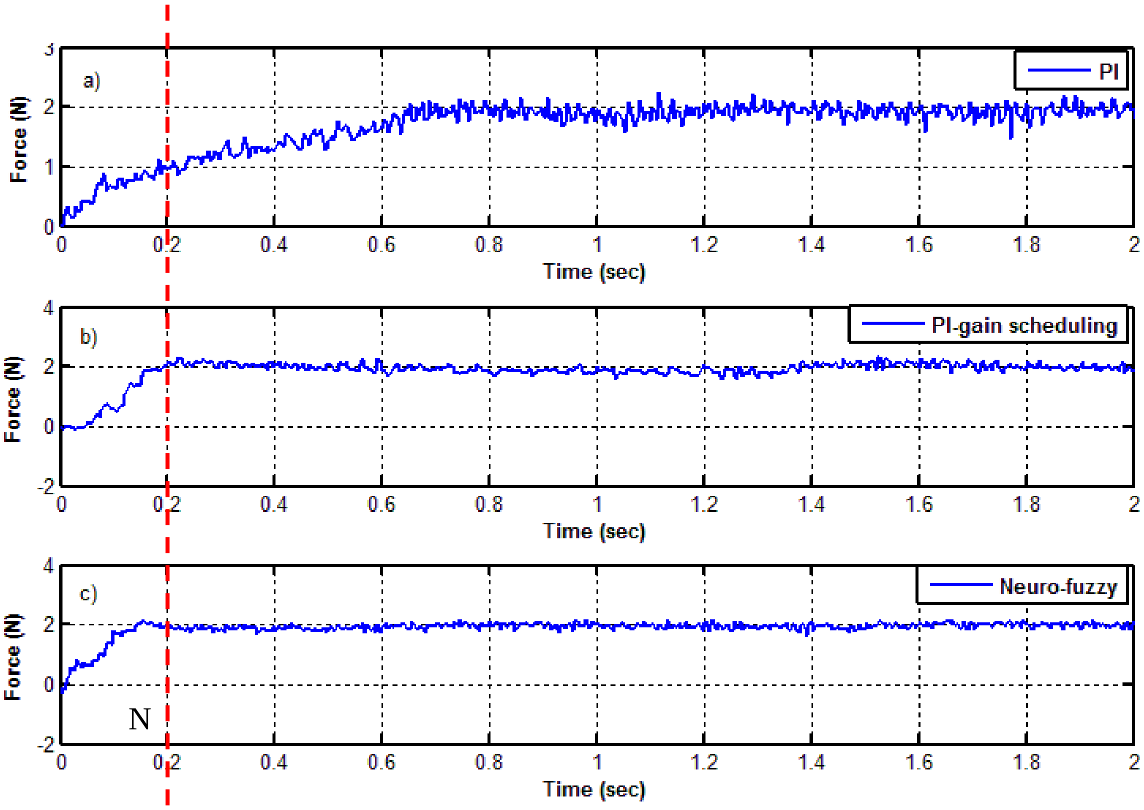


Figure 7-20 Comparison of step response of (a) PI, (b) PI gain scheduling and (c) Neuro-Fuzzy force controller, k_e = 50N/mm

Implementing PIGS technique further, in Figure 7-20(b) shows an improved force response with even less delay time, rise time and settling time than Figure 7-20(a). The technique which is the form of a look-up table, containing all the possible variations in force response error with matching P and I gains is shown effectively improved the system response. This is shown to give a considerably improved force response ITAE index of 313.97 N.s (Table 7-6).

Table 7-6 PI, PIGS and NF force controller performance evaluation of k_e = 50N/mm

Controller	t_d (s)	t_r (s)	t_s (s)	IAE (N)	ITAE (N.s)
PI	0.2	0.65	0.72	222.39	444.77
PIGS	0.12	0.16	0.26	156.99	313.97
NF	0.07	0.12	0.18	111.77	223.54

Implementing Neuro-Fuzzy technique still further, gives the results of Figure 7-20(c) which shows further improvement in force response over that of Figure 7-20(a) and Figure 7-20(b). The intelligent prediction algorithm of Neuro-Fuzzy predicts the stiffness changes and compensates the force response error, thus accelerate the system response as indicated by performance evaluation of Table 7-6. As can be seen from Figure 7-20(c) and Table 7-6, the delay time, rise and settling time are greatly improved by approximately 50%, 25% and 30% respectively compared to the PIGS technique. Furthermore, the force signal ripple is far much reduced as the lowest ITAE (223.54 N.s) was verified. These results study suggest that the NF can be employed to enhance the conventional method of force controller in varying environment stiffness. Undesirable effect can be minimized by online tuning procedure using combination of designers' empirical, analytical and intuitive decision making approaches.

As tuning of the PI gain scheduling process involved a lengthy trial and error process, the Neuro-Fuzzy controller has the advantage on processing the input and output information and refining the membership functions in a systematic way. With the availability of learning algorithms, a wider range of controller parameter estimation is expected compared to the PI gain scheduling method. The Neuro-Fuzzy controller switches the feedback gains smoothly unlike the gain scheduler. Furthermore, the proposed approach is applicable for use on uncertain environments and does not require any actual pre-learning process. Other researchers have anticipated environment detection and stiffness estimation techniques [Kiguchi and Fukuda, 1997; Lin and Huang, 1997; Ow, 1997; Lin and Huang, 1998; Seul and Hsia, 1998; Kiguchi and Fukuda, 1999].

So far, the Neuro-Fuzzy Force Controller (NFFC) has been established to overcome unknown and varying environment stiffness problem. The combination of fuzzy control and neural network system provides an effective approach for force control on fixed environment. However, if the environment position is continuously moving the controller has to be enhanced to adapt the situation. In order to resolve this problem, an online controller adjustment method is proposed which adjusts the NFFC immediately by multiplying the adjustment factor coefficients according to environment dynamic changes particularly due to the uncontrollable environment movement. The method is very effective for the robot force controller to deal with unknown environment parameters using a limited number of control rules. The Neuro-Fuzzy learning and adaption mechanism algorithm based on model reference adaptive control is applied as described in following section 7.8.

7.8 Design of Adaptive Neuro-Fuzzy Force Controller based on MRAC

The Adaptive Neuro-Fuzzy Force Controller (ANFFC) structure is similar to that of a classical model reference adaptive control (MRAC) method as shown in Figure 7-21. The general idea of MRAC is to create a ‘parameter adjustment’ algorithm based on the model error ε to change the system response to match a desired model. The output of the parameter adjustment algorithm δ approaches unity as the system response tends to desired response. There are many different approaches for designing MRAC controllers, the gradient method (MIT rule), least squares and those based on the strictly-positive-real (SPR) Lyapunov design approach [Åström and Wittenmark, 1995; Ioannou and Sun, 1996].

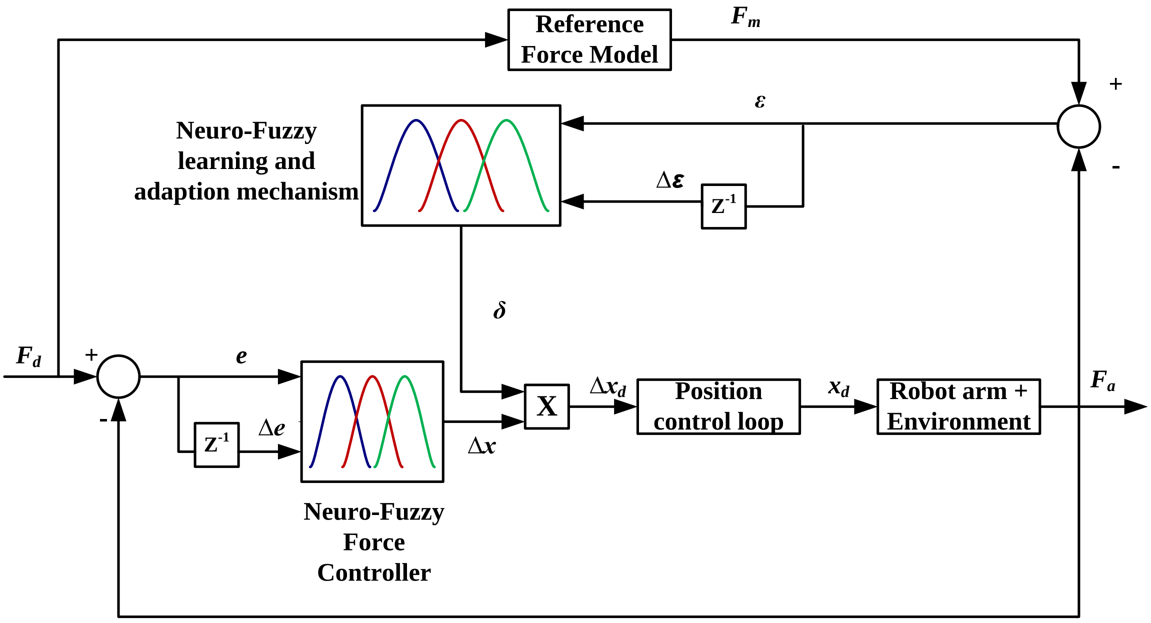


Figure 7-21 The Adaptive Neuro-Fuzzy Force Controller (ANFFC) structure

The implementation of ANFFC is to compensate for any unknown or significant change in environment parameters that may lead to overshoot and undershoot error responses. This method behaves in a similar way to a human does, as we do not apply every control rule for particular situation but moderately adjusts the existing Neuro-Fuzzy Force Controller (NFFC) corresponding to the actual situation. The Neuro-Fuzzy concept is again applied to determine and generate the effective parameter adjustment algorithm.

7.8.1 Adjustment algorithms of Neuro-Fuzzy Force Controller

The ANFFC consists of three main components: a reference model describing the desired behaviour of the force control scheme, a main Neuro-Fuzzy force controller (NFFC) that determines the adjustment to the position control loop and Neuro-Fuzzy learning and adaption mechanism (NFLA) that modifies the NFFC output.

The reference model is chosen to generate the whole system desired behaviour in response to a force trajectory. The model can be designed as any linear or nonlinear response, for example as a second-order linear system as given by equation (7-12).

$$\frac{F_m}{F_d} = \frac{1}{s^2 + 2\zeta\omega s + \omega^2} \tag{7-12}$$

The parameters ζ and ω represent the desired damping ratio and natural frequency respectively. In this research, the reference model is generated using the MATLAB® System Identification Toolbox™ software as discussed in section 5.3. Prior to that, a step force response test was conducted to identify the input-output data set of desired behaviour of force control system. The desired behaviour criteria are 0.2s rise time, overshoot of 10% with 5% steady-state error

The input-output data was used to estimate a second-order linear system (7-12) parameters and the model output is validated by using model-output plot where the identified model (7-13) accurately fit the measured data of ~90%.

$$\frac{F_m}{F_d} = \frac{1}{s^2 + 7s + 100} \tag{7-13}$$

where ζ and ω are 0.35 and 10 respectively.

In order to implement this equation for discrete-time implementation, a bilinear (Tustin) transformation available in MATLAB® was used to convert the estimated continuous-time model. The discretized model was compare to the continuous-time model for validation before implementing the model on real system.

The purpose of the NFLA is to learn and adapt to any changes in the environmental parameters and subsequently modify the NFFC output accordingly so that the overall system response is as close as possible to the desired reference model. The NFLA inputs are the difference between the reference model output and the measured contact force ε and the rate of error $\Delta\varepsilon$. Its output is the parameter of factor adjustment δ which is used to modify the output of NFFC. The method is very useful in compensating any disturbance caused by environment position (moving object) which is a key requirement for the TMS robotic system. The Neuro–Fuzzy approach was applied to evaluate the degree of influence of the undesired response, and consequently, the adjustment algorithm generates a larger factor adjustment parameter for the larger undesired oscillation and unexpected overshoot/undershoot error. Similarly a smaller value is generated if the measured force is closed to the desired response. The following section discusses the design of the adjustment algorithm in detail.

7.8.2 Design of Neuro-Fuzzy Learning and Adaption mechanism

Similar to the approach discussed in section 7.6, the rule-base is created as follows;

If $\varepsilon(k)$ is ε_i and $\Delta\varepsilon(k)$ is $\Delta\varepsilon_i$, then output is δi

To begin with, the environment movement can be classified in terms of the sign of the model error ε:

1) **Negative** if the object moves towards the Stäubli robot

2) **Positive** if the object moves away from the Stäubli robot

Similarly the force controller behaviour can be defined by the rate of error sign as summarized in Table 7-7.

Table 7-7 A rule-base parameter determination

Δε sign	**ε sign**	**Force controller response**	**Adjustment**
Zero	Zero/Positive/Negative	Satisfactory rate	No or small adjustment
Positive	Positive	Slow	Large adjustment
	Negative	Satisfactory rate	Small adjustment
Negative	Positive	Satisfactory rate	Small adjustment
	Negative	Slow	Large adjustment

A rule base is designed to reflect the following cases;

Case 1 No adjustment

This is the case in which the deviation of model error and output of the NFFC are closed to zero. In this case the actual force response is closely tracking the desired reference model, thus no adjustment is required, ($\delta = 1$).

Case 2 Small adjustment

Small adjustments are required in any of the conditions;

a) ε is positive and Δε is small
b) ε is small and Δε is positive
c) ε is negative and Δε is small
d) ε is small and Δε is negative

The above conditions suggest that the object is moving away from (conditions a and b) or towards (conditions c and d) the Stäubli robot, but the system is responding correctly since the model error ε and derivative of error $\Delta\varepsilon$ value is small.

To explain this, consider condition (a) which indicates that the actual force is lower than desired force but the system will respond correctly as the rate of error values tends to be zero. This condition only needs a small adjustment in the positive direction and consequently changes the NFFC output to increase incremental position Δx_d. Similar arguments can be applied for other conditions (b), (c) and (d).

Case 3 Large adjustment

A large adjustment is needed if the model error ε and derivative of error $\Delta\varepsilon$ is in the same sign. Consider the following rule;

If $\varepsilon(k)$ is positive and $\Delta\varepsilon(k)$ is positive

The rule indicates that the object is moving away and the controller is responding too slowly, thus a large adjustment is needed to correct the situation. Similarly, if the object is moving towards, both are negative, a large adjustment in negative direction is necessary.

Note that the above cases are intended to provide only a general guideline for the selection of the rule, in order to improve the controller performance the input variables is normalized into seven membership functions which are negative big (**NB**), negative medium (**NM**), negative small (**NS**), Zero (**ZE**), positive small (**PS**), positive medium (**PM**) and positive big (**PB**).

As mentioned earlier, there are no theories or established methods to define and adjust the membership functions. It is necessary to carefully experiment and determine the optimum results. In this work, the fuzzy inference system is again tuned using MATLAB® ANFIS toolbox to compensate the object movement in order to maintain the contact force. The ranges of different input variables to the fuzzy sets were divided carefully to cover all defined conditions in Table 7-7 and provide the system with greater flexibility to make the best decisions. These processes using a systematic trial and error approach, based on gain scheduling method in order to find appropriate and suitable factor parameters.

The training data for the NFLA is initially expecting object movements obtained from the pre-experiment of factor scheduling. With this training data, the NFLA is trained offline using the least square and back-propagation learning algorithm to output a number of factors according to the object movement. Test observations brought about the mechanism tuning by changing the rules, adjusting the membership functions of the inputs, and re-examining the response of the ANFFC. Once the procedure had been conducted several times, stable system behaviour was achieved. The design of ANFFC can be summarized as following Figure 7-22;

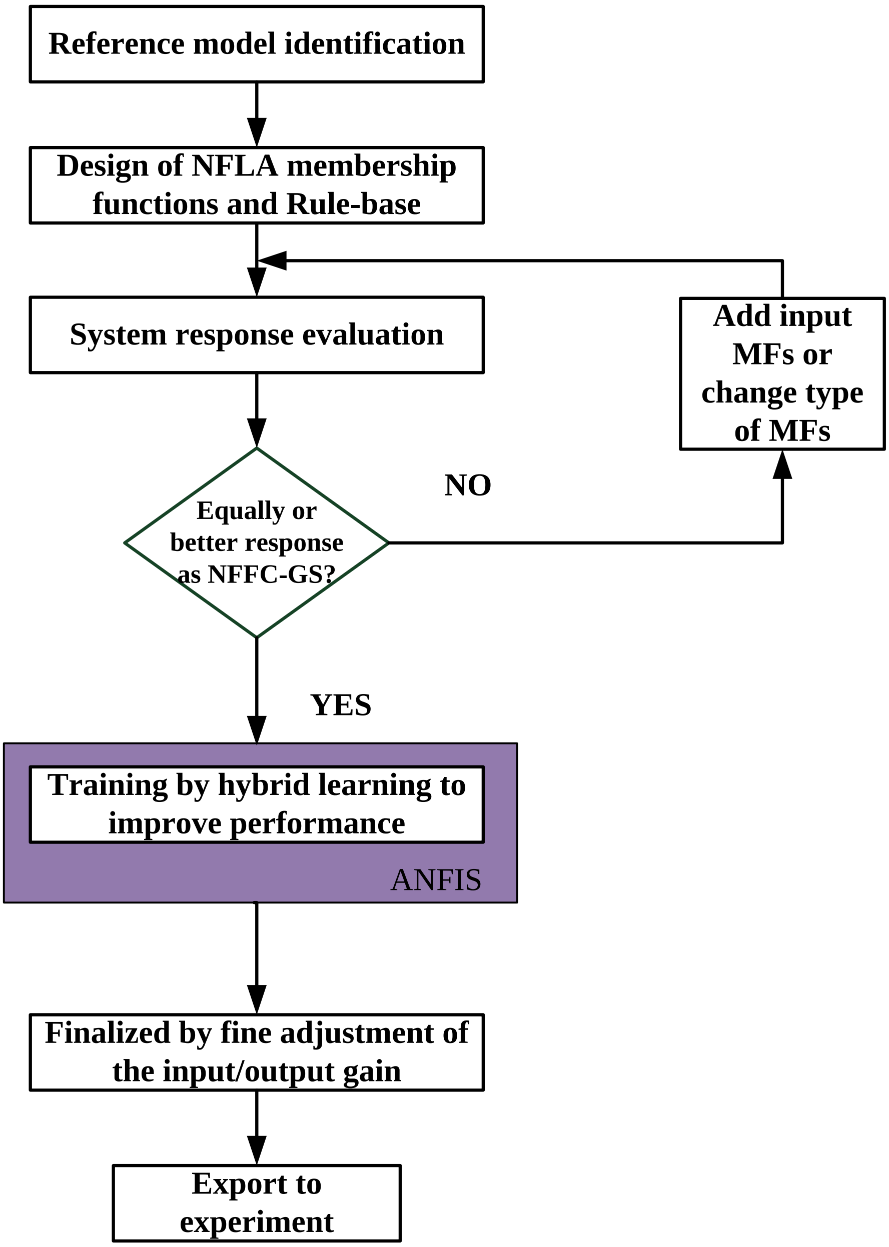

Reference model identification
Design of NFLA membership functions and Rule-base
System response evaluation
Add input MFs or change type of MFs
NO
Equally or better response as NFFC-GS?
YES
Training by hybrid learning to improve performance
ANFIS
Finalized by fine adjustment of the input/output gain
Export to experiment

Figure 7-22 Design of ANFFC

Apart from improving the robot response, there are other factors which need to be considered. In particular, upon first contact with the environment, the system will sense a rapidly increased in force error value. Subsequently, a high value of factor adjustment δ is set by NFLA mechanism. This situation potentially creates a large impact force due to a high value of incremental position, and results in a repetitive bouncing action thus introduce instability to the system. To prevent this instability the NFLA output is kept constant for a period of one second on contact to allow the system to settle. Once this has passed, the NFLA is changed accordingly. The following section accesses the performance of the proposed adaptive Neuro-Fuzzy force controller.

7.8.3 Preliminary test of Adaptive Neuro-Fuzzy Force Controller

Thus far, the response behaviour of the proposed control method has been tested on a static environment. This section describes how the force controller is able to accommodate a dynamic environment which is not stationary; namely a sequence of ramp inputs with/without parameter adjustment algorithms. Figure 7-23 illustrates the schematic diagram of ANNFC test set up. The Stäubli robot is first moved down at an approximate 10 mm/sec until it makes contact with the ball. Once contact is made, the force control law is activated in which the robot was commanded to maintain 5N contact force in the *z* axis direction. Then, after approximately 7 seconds the PUMA robot is moved about 10mm in towards direction (from position B to A) to the Stäubli robot with a speed approximately 2mm/sec. Another PUMA robot movement (from position A to B) was applied for the same travel distance. Finally, the Stäubli robot is withdrawn from the contact environment. The sequence was repeated 5 times and the force response (*z* axis) is recorded throughout the test. Note that, neither environment stiffness nor its position is known to the robot controller.

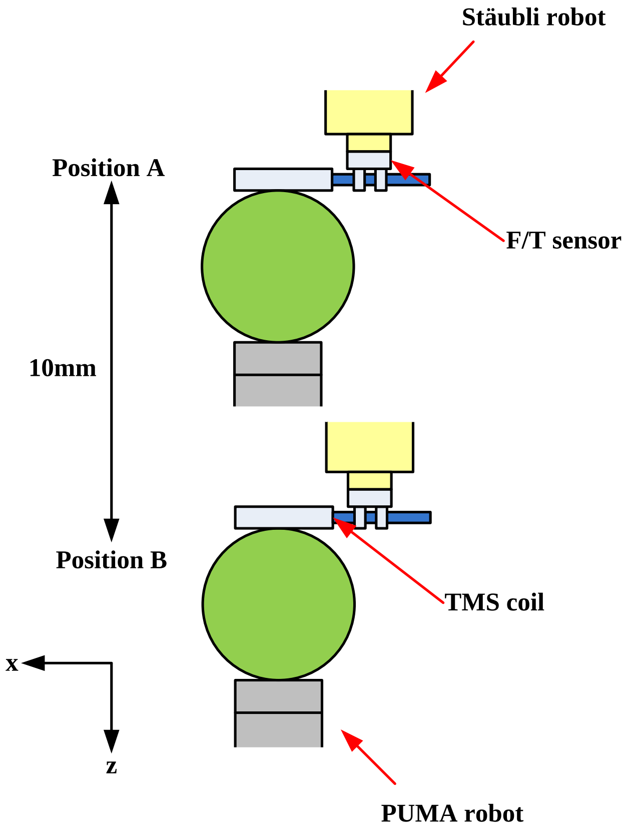


Figure 7-23 ANFFC test schematic diagram

7.8.3.1 *Test results and discussion*

Figure 7-24 shows the results of force and position (along *z* axis) during this test. There was a significant difference between the conventional PI and the ANFFC controllers. It is apparent from the data in Figure 7-24(a), that the conventional PI force controller drives the manipulator into an unstable limit cycle. The problem can be attributed to the rapid changes in movement (either robot or environment) that leads to impulsive forces on the system. The form of the signal shows the PI controller was trying to compensate the impulsive force, but failed to do so. In this condition, it is desirable if the Stäubli robot can moves at velocity equal to the environment speed. Consider the situation in which the environment moves towards the robot very rapidly, the large force error occurs before the robot controller can react to it due to small calculated incremental position. This results in the robot springing backward rapidly. Similar pattern is observed during away movement, if the environment movement speed is higher than resultant incremental position, the robot momentarily leaving contact with the environment. This inevitably results in repetitive bouncing action until the environment stop moving. Clearly, the dynamics of the force-controlled system are now considered to be extremely complex with many unknown independent factors. Thus the task of maintaining force response, as position changing with the conventional PI controller only is impossible.

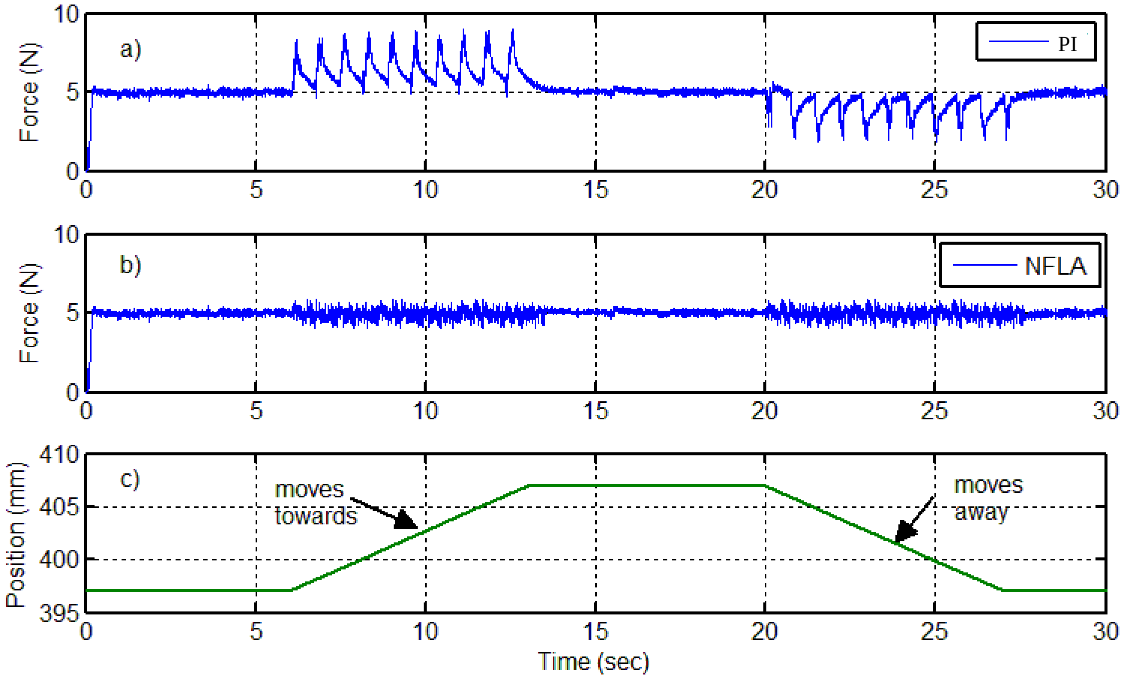


Figure 7-24 Test results of force response of (a) PI and (b) NFLA, and (c) Stäubli robot position

For this reason, the problem can be reduced to some extent by ANFFC robot controller implementation. Although it shows a great improvement in terms of repetitive intermittent contact, the signal is fluctuated about 16% from the desired force which in the author opinion is acceptable. A force response data during environment movement was extracted and the performance of both PI and ANFFC were compared by ITAE. The ITAE index for PI and ANFFC are 2623N.s and 506N.s respectively. As expected the NFLA mechanism of ANFFC provides a good tracking performance by improvement of 81%. In this implementation, the desired incremental position resulting from the main loop NF force controller is magnified by the NFLA output, and subsequently increased the Stäubli robot speed whilst simultaneously maintaining a desired contact force with the environment.

However, bearing in mind there is always a limit for the NFLA output, for instance, when the contact is lost the system sets the higher output (high velocity) that can create limit cycle. This results in intermittent contact that may harm the subject. In this case, the NFLA is need to finely tuned to minimize undesirable force response in sensed that a robot can only follow a certain identified range value of environment speed. Further experimental work, analysis and limitation on this topic will be discussed in the following chapter.

Extensive prior study (Chapter 5 and 6) on force control has delineated the main factor of instabilities that can arise when controlling a desired contact force. It is clear that there is a trade-off between proportional force controller gain *Kp* and environment stiffness *Ke*. An increase in *Ke* decreases the *Kp*, subsequently decrease the velocity of the robot v_S. Another factor that limits the robot velocity is the force error F_e itself as the lower force error lowers the robot velocity as velocity proportional to force error. The velocity limitation also depends on ALTER function performance to handle translational position every sampling period. The limitation of robot velocity during contact condition can be defined as following equations;

$$v_s = \begin{cases} min, & K_e \cong K_{e(\max)} \\ max, & K_e \cong K_{e(\min)} \end{cases} \tag{7-14}$$

$$v_s = \begin{cases} slow, & F_e \cong small \\ fast, & F_e \cong large \end{cases} \tag{7-15}$$

$$v_s = \begin{cases} min, & v_{ALTER} \cong v_{ALTER(\text{min})} \\ max, & v_{ALTER} \cong v_{ALTER(\text{max})} \end{cases} \tag{7-16}$$

7.9 Summary

In this chapter, the PI gain scheduling and Neuro-Fuzzy methods were successfully implemented in TMS robotic system and demonstrated improvement for the control system. Experimental results have shown the effectiveness of the proposed controller particularly for unknown/variable environment stiffness as well as in condition when the environment parameters and position are unknown and change substantially. In addition, the proposed method also employs a learning and adaptive mechanism to adjust a main loop Neuro-Fuzzy force controller online. The purpose of this mechanism is to learn the environment and adapt the controller accordingly. Adaption consists of comparing the model reference error and using the derivative of error as the input to the Neuro-Fuzzy algorithm. Any undesired response is immediately compensated with a suitable adjustment parameter according to the response behaviour. The evaluation of the proposed method will be demonstrated and discussed in detail in the following chapter.

CHAPTER 8

IMPLEMENTATION AND EVALUATION OF PROPOSED FORCE-CONTROLLED TMS ROBOTIC SYSTEM

The previous chapter has described how the Stäubli robot system can be programmed to perform force-controlled tasks. As discussed in Section 3.1, in the context of force control strategy, the TMS procedure comprises of two phases, namely preoperative phase (free space movement) and an intraoperative phase, which involves maintaining a force normal to the subject's head (constrained space movement). Both phases will be described in detail in Section 8.1.

In the following sections, the experimental work has been conducted using the proposed force-controlled TMS robotic system as discussed in Chapter 4. For the safety reason, a ball mounted on a six-degree of freedom PUMA 560 manipulator end effector was used to simulate the human head and its motion. In Section 8.2, the circular tracking test was developed to demonstrate the tracking ability of *tele-operation* mode. Judgements for the proposed force-controlled system are made by considering three important criteria for the successful implementation of the system; stability, transparency of operation and ability to minimise operator fatigue.

Early experimental works showed that force control stability presented major problems and as a consequence this topic is extensively investigated. In addition to that, in TMS application this is becoming more problematic as there is always an involuntary subject's head movement occurred during the TMS procedure. A conventional force controller was found not suitable to resolve this problem, thus a Neuro-Fuzzy Learning and Adaption (NFLA) mechanism was proposed. Section 8.3 will describe the NFLA mechanism implementation to the force-controlled TMS robotic system. A specific analysis of the transient force response has also been discussed.

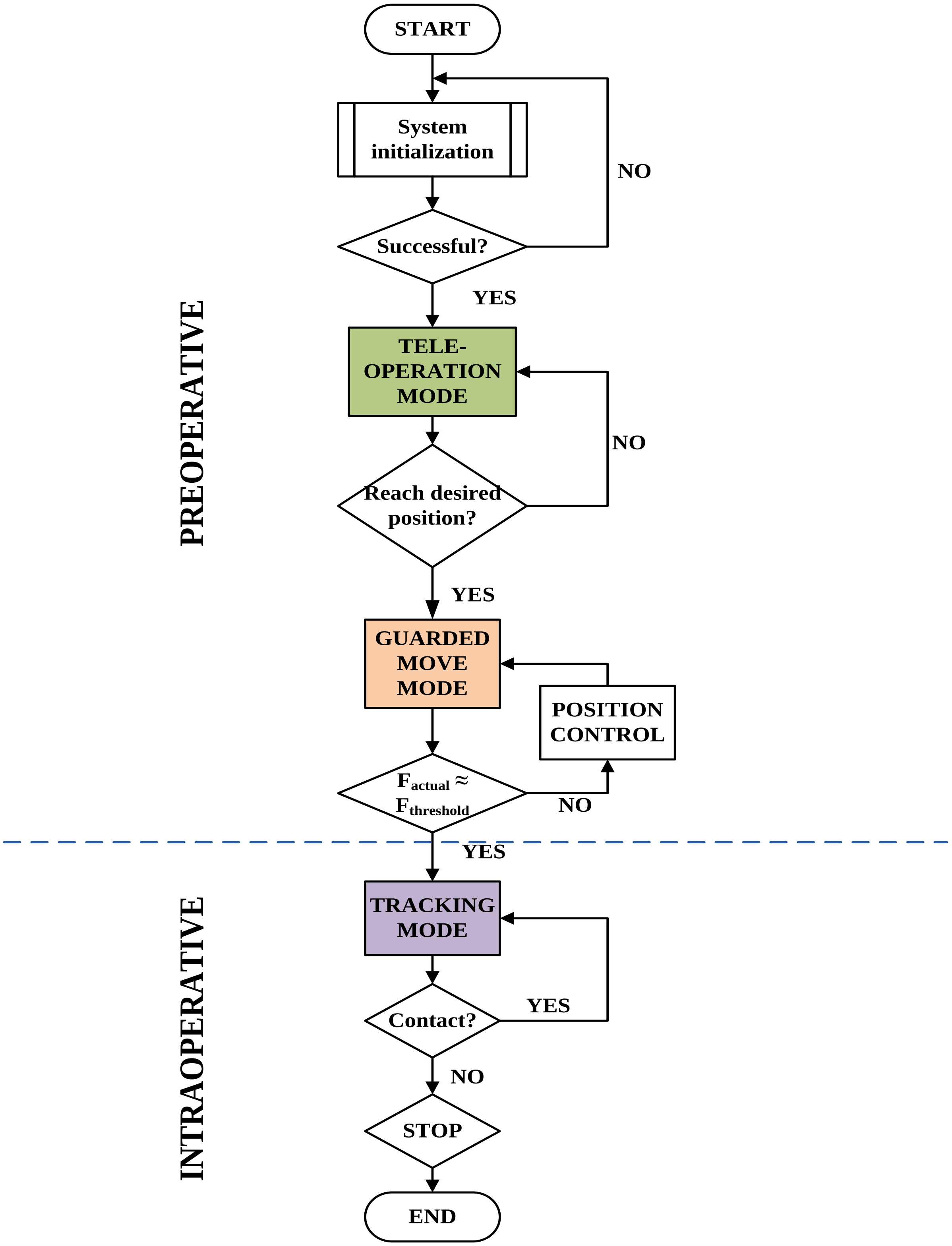


Figure 8-1 TMS force-controlled robotic system procedure

8.1 Force-Controlled TMSRobotic System Procedure

As described in the Chapter 3, the procedure of force-controlled TMS robotic system can be divided into two phases: preoperative and intraoperative phases as shown in Figure 8-1.

Preoperative phase: This phase comprises of two modes namely '*tele-operation*' and '*guarded move*'. The motion begins with *tele-operation* mode, which enables the operator to make gross spatial maneuvers of the robot arm to locate the target position (point of contact). Figure 8-2 shows the block diagram of the *tele-operation* control scheme.

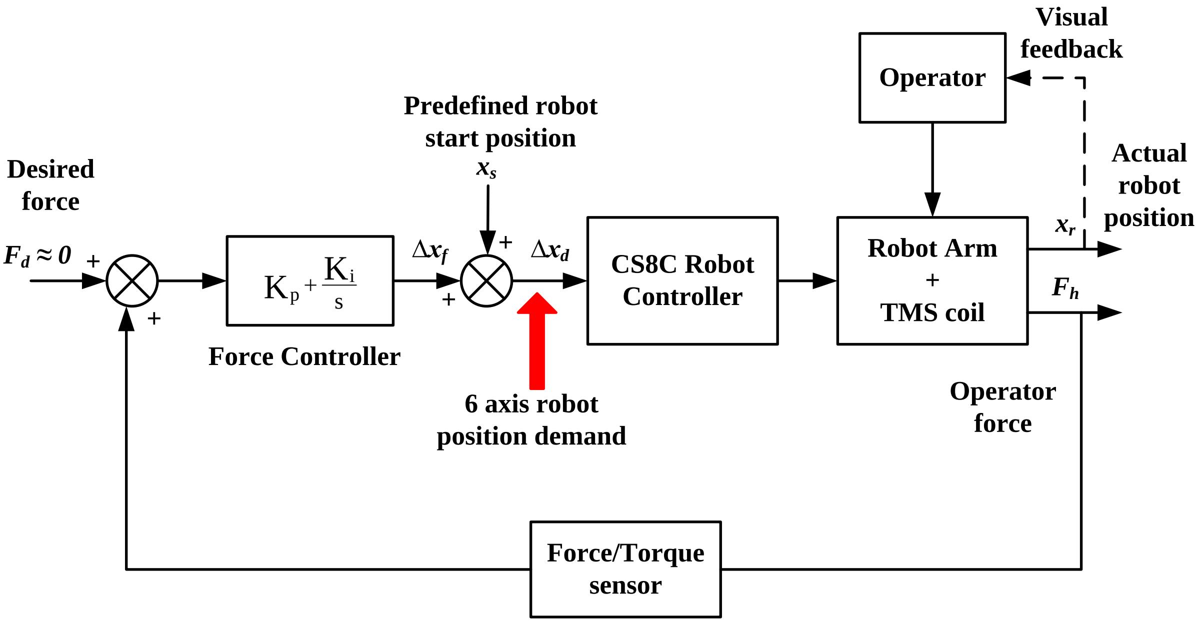


Figure 8-2 Block diagram of *tele-operation* mode control scheme

In this control mode, the force applied by the operator's hand is measuring using the ATI Mini40 F/T sensor mounted on the robot arm end-effector. The force/torque data is then routed to the PI force controller as shown in the following equation.

$$\Delta x_f = \left(K_p + \frac{K_i}{s}\right) F_h \tag{8-1}$$

This force controller generates the incremental position demand Δx_f for the robot arm. Then, the robot will move at a velocity proportional to the applied force F_h. The implementation of an external force feedback control loop provides a form of kinaesthetic feedback to the operator since the operator is able to freely move the robot. Note that the presence of the visual feedback enables the operator to adjust his/her applied force F_h as task progresses.

Once the operator defines the target position, the control system performs a '*guarded move'* in the specified direction. The *guarded move* consists of a sequential **'Move-To'** and **'Move-To-Contact'** routine. The **'Move-To'** routine moves the coil from its current position to a position specified by the operator. Then, the **'Move-To-Contact'** routine, as the name suggests, moves the coil into contact at constant low velocity until contact is made with the subject's head, which is detected when the force threshold is sensed. For both routines the robot follows a 5th polynomial velocity profile, in which the direction of movement is specified along the normal direction to the subject's head surface.

Intraoperative phase: Once the preoperative phase has been successfully carried out, the *tracking* mode is activated to maintain a desired low contact force. The contact force between the coil and the subject's head is maintained using the force feedback loop as discussed in Chapter 6 and 7. Once the TMS treatment is completed, the operator can stop the *tracking* mode with the aid of the graphical user interface and fully withdraw the coil above the contact surface.

8.2 Performance Evaluation of Preoperative Phase

8.2.1 Tele-operation tracking ability test

8.2.1.1 *Task Description*

A circular tracking test was used to demonstrate the ability of *tele-operation* mode which required the operator to track a circle of approximately 100mm diameter [Burn, 1993]. This test is also able to illustrate the effects of varying the control gain close to the maximum value for stable operation, as a high gain is preferable in order to maximise the 'transparency' of the control task. The objective of this test was to move the TMS coil handle in an XY plane. The circle was drawn on a white template and mounted on the fixed platform which was positioned within the robot workspace. A small pointer was fixed to the TMS coil to provide a visual indication to the operator during the test. The robot was positioned such that the pointer was a short distance (≈35mm) above the template. A schematic diagram of the test arrangement is shown in Figure 8-3.

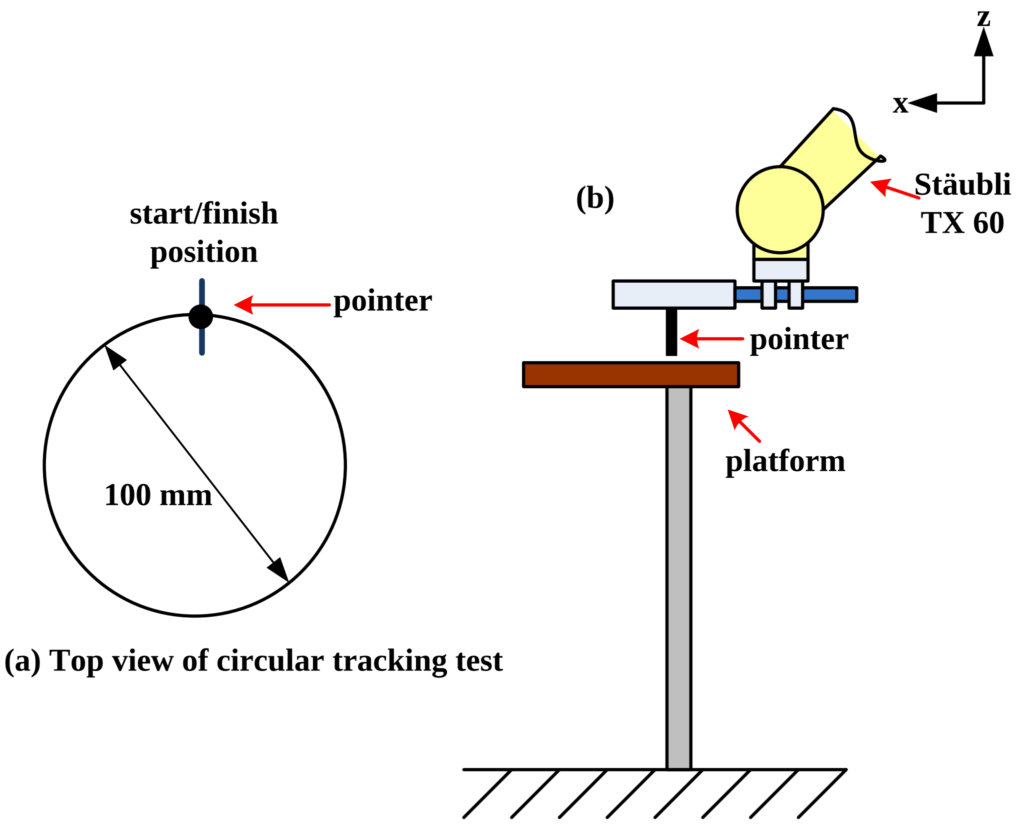


Figure 8-3 Test set up for circular tracking test

The test sequence can be summarized as follows;

1. The XYZ translational world axes were activated and all other axes isolated in main software program. Prior to running the real-time ALTER loop implementation, an approach procedure was initiated in which the robot is commanded to move to a predefined starting position on the circular track.
2. A test was initiated by pressing the ENTER key of the PC keyboard, to enable the real-time control loop. The operator was then required to track the circular path in a clockwise direction. A test was completed when a full revolution had

been achieved, and the program aborted by pressing any key of the PC keyboard or robot teach pendant.

When operating in *tele-operation* mode, three of the most important criteria for the successful implementation of the system are its stability, its transparency of operation and its ability to minimise operator fatigue. Consider the resultant incremental position given by

$$\Delta x_d[i] = (K_p + \frac{K_i}{s}) * F_h[i] \tag{8-2}$$

where K_p and K_i are proportional and integral gains and F_h is input force for i'th axis (x, y or z).

Choosing the optimum empirical control gains is a cautious exercise. A high gain is preferable as its implementation will increase the transparency of the operation. However the gain can only be increased to a certain upper limit; if the gain is set too large the system becomes unstable which could easily damage the robot arm, sensor, TMS coil and subject's head. Potentially, this effect is more visible by the natural tendency to grip the handle more tightly on the onset of instability, which effectively increases the stiffness of the system. To avoid this, the gain is gradually decreased but a low value of the control gain results in a slow, sluggish response as a higher force is required to generate the desired robot arm velocity. This leads to increased operator fatigue.

8.2.1.2 *Test results and discussion*

To confirm the significance of the control gains, a similar test was repeated for different proportional gains and the integral gain was set at K_i = 0.005, which is the optimum integral gain for desired steady-state response. The test was repeated 5 times in a random order in which XYZ position data was recorded in real-time. The mean absolute positional error (MAPE) between the pointer and the circle was calculated and results obtained for the test at gains ranging between 0.001 and 0.02 are given in Table 8-1 and graphically represented by Figure 8-4. The manoeuvrability evaluation was provided by mean radial force (MRF) measurement as shown Table 8-1.

Table 8-1 Results of circular tracking tests

Proportional Gain, Kp	Mean Position Error (mm)	Mean Radial Force (N)
0.001	0.94	19.80
0.0025	0.90	7.11
0.005	0.77	4.32
0.0075	0.88	3.36
0.01	0.53	2.90
0.012	0.57	3.16
0.015	0.65	3.11
0.02	1.13	3.14

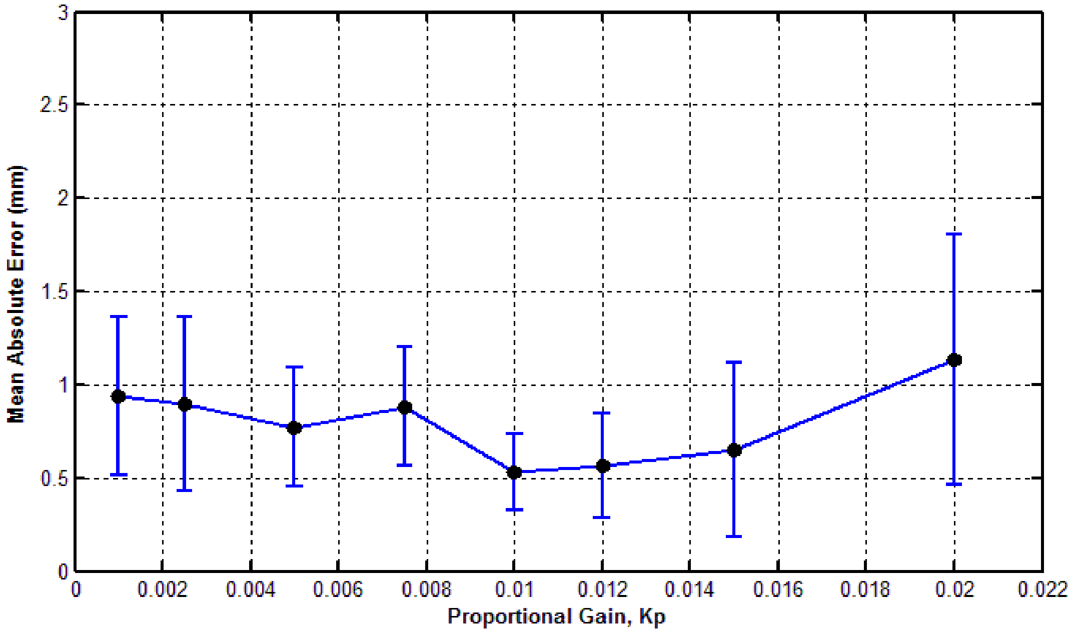
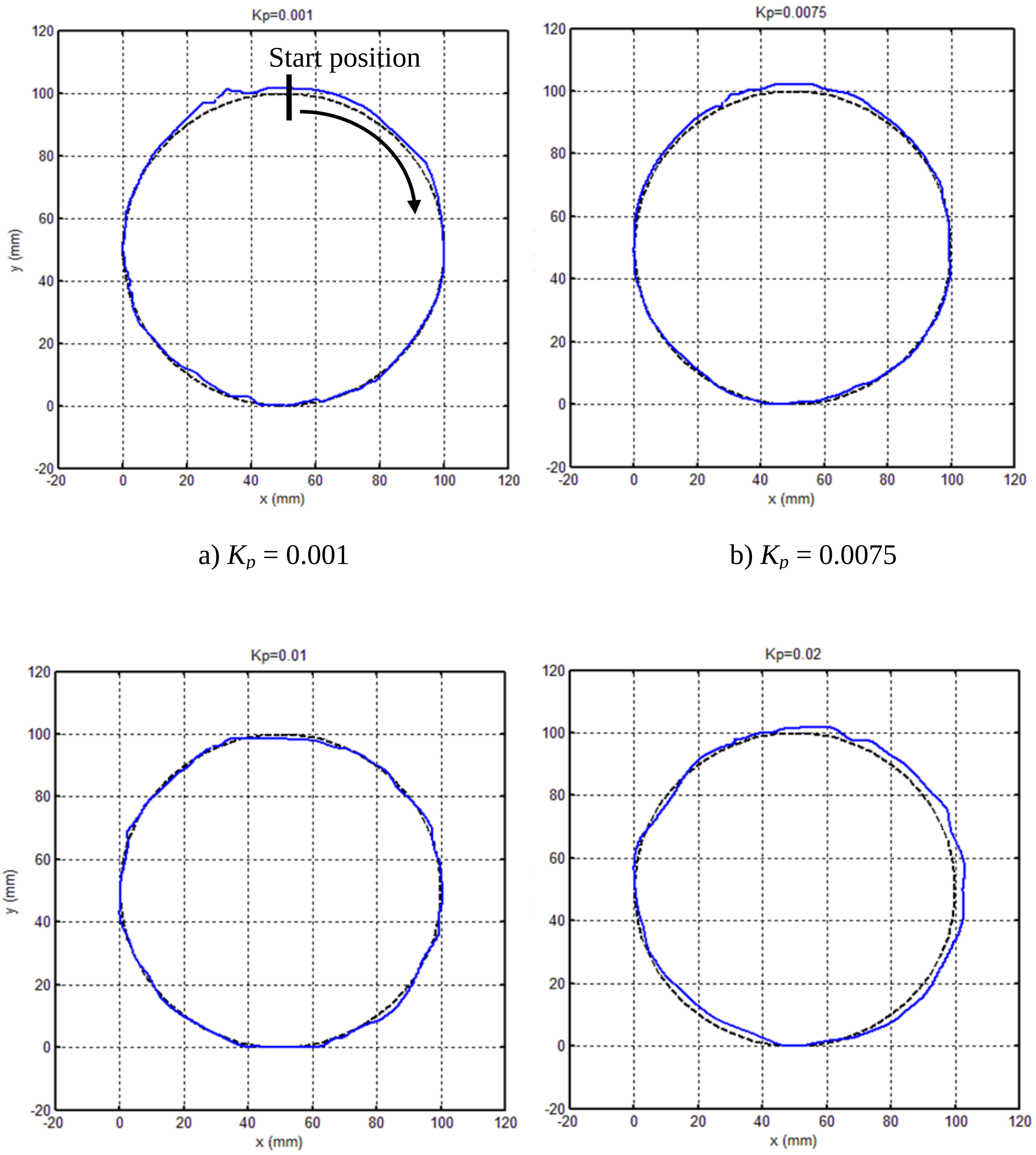


Figure 8-4 Results of circular tracking test, mean absolute error versus proportional gain.

The results of Figure 8-4 shows that gains ranging between 0.0005 and 0.015 give the best performance as indicated by low mean value of positional errors (< 0.7 mm). This is confirmed by representative plots of Figure 8-5(b) and (c), which shows the minimum tracking error relative to the dotted ‘ideal’ circle. The controller with proportional gain of 0.01 is shown to have the lowest value of mean position error with minimum standard deviation interval as represented by the error bar. At this point, a minimum mean radial force was recorded in which determines the optimum gain for the best robot manoeuvrability.

Increasing the gain further is accompanied by an increase in the mean position error and standard deviation interval, indicating a decrease in the effectiveness of tracking task. As can be seen in Figure 8-4, the largest mean error was recorded for the controller gain of 0.02 with the largest positional error of more than 1mm. At this point, it was found quite difficult to manoeuvre the robot arm as increase in proportional gain can cause the system to be too sensitive. A small increase in gripping force at the handle was enough to stimulate the movement and subsequently make the system oscillate. The system was becoming close to instability and further increasing of gain will degrade the system performance.

Reducing K_p below than 0.005 made the system become considerably sluggish and difficult to manoeuvre as a higher force were required to perform the task which in the author’s opinion is more tiring to operate. As can be seen in Table 8-1, reducing the proportional gain is accompanied by an increase in applied force. Furthermore, a mean position error is also increased. Figure 8-4(a) shows the results of performing the task at a gain of 0.001. The results indicate that a large position error was recorded at beginning of the tracking task as relatively large force was applied to the handle to initiate the robot to move. The degradation in tracking performance can clearly be seen, with larger position errors at gains lower than 0.005 and higher than 0.015.



a) K_p = 0.001

b) K_p = 0.0075

c) K_p = 0.01

d) K_p = 0.02

Figure 8-5 Circular tracking test results with varying K_p

In addition to *x* and *y* positions, movement in the *z* axis were also recorded. Plot of *z* position vs. time for three different gains is shown in Figure 8-6. The results demonstrate the arbitrarily vertical robot motion which occurred whilst moving in a horizontal plane, in which interestingly demonstrates how closely they followed each other. In Figure 8-6, an oscillatory response is observable at the highest gain (K_p = 0.02) at the start point of circular tracking task. This result again indicates the degradation in performance with the higher gain. At this point, it is recommended to apply a ‘light’ grip of the handle to prevent any unwanted movement of the robot. These results demonstrating that three of the most important criteria for the successful implementation of the system which are its stability, its transparency of operation and its ability to minimise operator fatigue is relied on proportional gain. Satisfactory performance was achieved with the gain in range between 0.0075 and 0.015 with low operator applied force at approximately 3N.

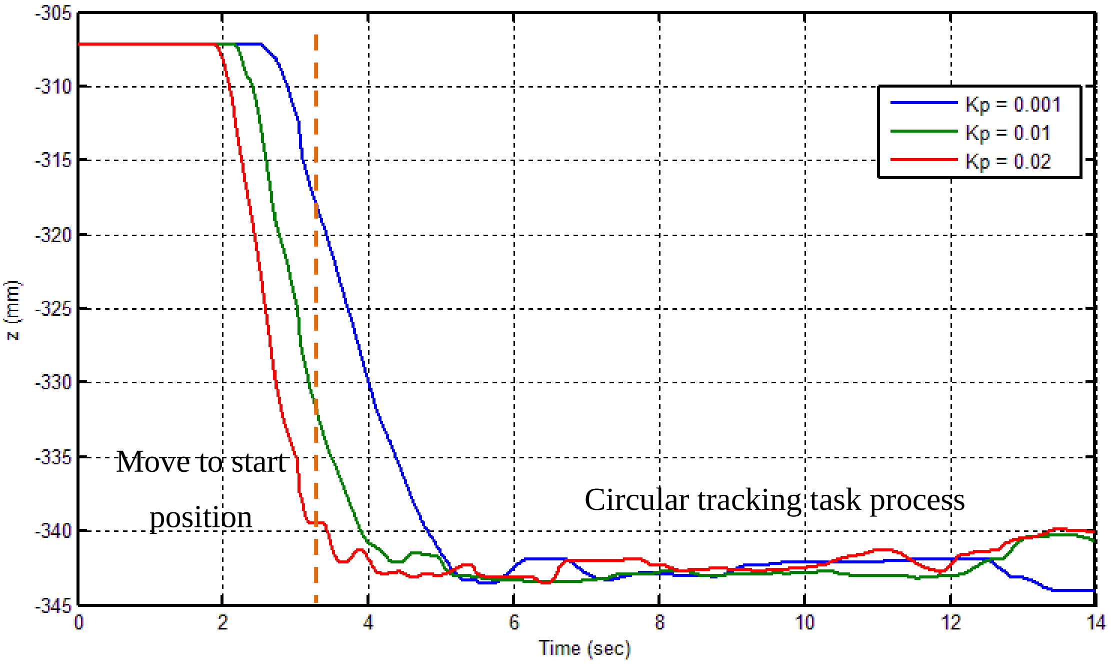


Figure 8-6 *z* axis position vs. time

8.3 Performance Evaluation of Intraoperative Phase

The need for robust force control in the TMS robotic system was discussed in previous chapters. Stability of force control was found to be the main problem and as a consequence the parameter affecting force response has been extensively investigated. Initially, in Chapter 5, a general 1 degree of freedom model simulation study was conducted. This was followed by the implementation of the conventional force control on the real 6 degree of freedom robot system. The preliminary experimental results show that the environment stiffness is a main factor of the stability and robust force control. It was apparent that a specified control gain was designed for particular environment stiffness. However, it was found problematic in force-controlled TMS robotic system as subject's head stiffness varies from one person to other person and can be considered unknown/uncertainties. Therefore, Neuro-Fuzzy learning adjustment mechanism was introduced to resolve the problem. The design of the mechanism has been discussed in detail in Section 7.6.8. This section will discuss the system response due to the human's head movement. A ball mounted on a six degree of freedom PUMA 560 manipulator arm was used as a contact environment in the following tests to simulate subject's head.

8.3.1 Effect of environment position changing and algorithm implementation considerations

There are many issues associated with the proposed control method implementation as discussed in Chapters 5 and 6. In particular, there is a significant relationship between the force control response and environmental stiffness properties. Preliminary test results suggest that the control gain has to be adjusted to reflect variations in the environment stiffness in order to achieve the same system performance specification. A Neuro-Fuzzy gain tuning controller was implemented in an attempt to accommodate unknown stiffness environment.

The proposed Neuro-Fuzzy gain tuning controller has demonstrated a number of performance benefits in the execution of robot force control on static environment with unknown stiffness (see Chapter 7). In TMS application, the system is required to autonomously maintain prescribed low contact force on the environment surface despite involuntary environment movement. One major concern is the environment movement can initiate the system to become unstable as discussed in simulation chapter. Involuntary environment movement becomes major concerns and often, major limitation. The environment movement tends to introduce instability to the control system leading to uncontrollable robot behaviour.

To understand this, considering the following Figure 8-7 shows that the robot system is coupled to the ball in three different conditions; 1)upon contact, 2)stationary environment position and 3)environment position changing (moving object). The condition 1 (preoperative phase) has been discussed in detail in Section 8.2, thus this section is focused on the force control task during intraoperative phase (condition 2 and 3). The system is assumed stable at condition 2, as the coupling imposes the velocities of the Stäubli robot and environment at a contact point to be equal. The actual contact forces F_c acting between the coil and environment can be considered equal to the desired contact force F_d as shown in following equation

$$if\ \ \vartheta_e = \vartheta_r, \qquad then\ \ F_a = F_d = k_e \cdot \delta_{sp} \tag{8-3}$$

where υ_e is environment's velocity, υ_r is robot's velocity, k_e is environment's stiffness and δ_{sp} is displacement to maintain a desired contact force.

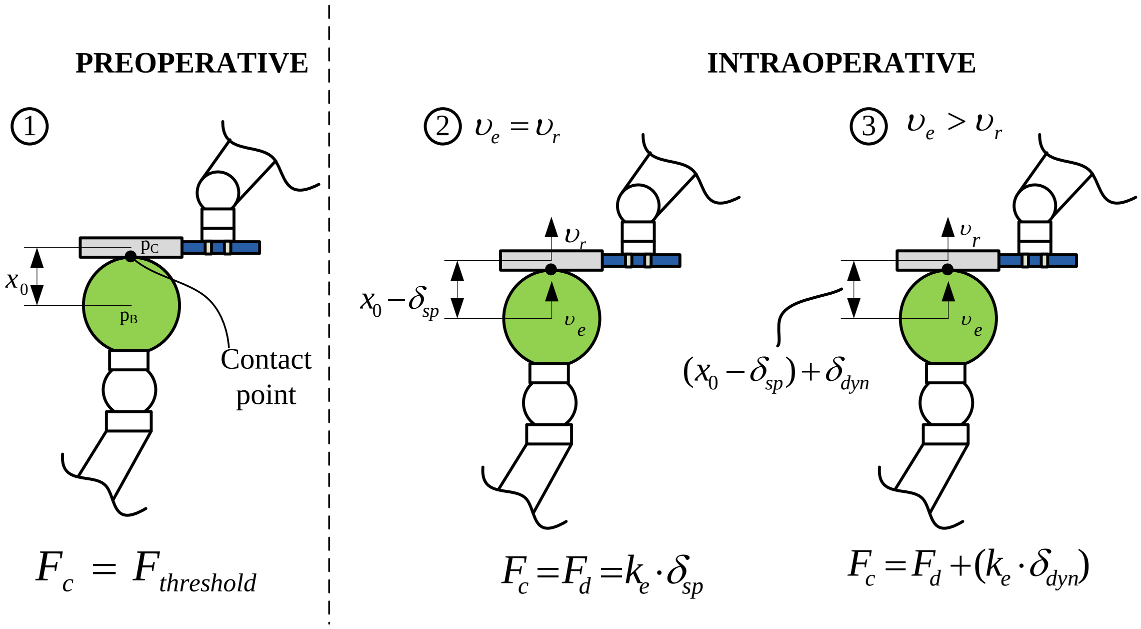

PREOPERATIVE
INTRAOPERATIVE
①
② $\upsilon_e = \upsilon_r$
③ $\upsilon_e > \upsilon_r$
p_C
x_0
p_B
Contact point
υ_r
$x_0 - \delta_{sp}$
υ_e
υ_r
$(x_0 - \delta_{sp}) + \delta_{dyn}$
υ_e
$F_c = F_{threshold}$
$F_c = F_d = k_e \cdot \delta_{sp}$
$F_c = F_d + (k_e \cdot \delta_{dyn})$

Figure 8-7 The coil and environment contact interaction diagram. x_0 is the distance between two arbitrary point in coil p_C and ball p_B at initial condition

However, as the environment moves for instance in towards direction to the coil (condition 3), the desired force is perturbed by the reaction force (F_R = $k_e.\delta_{dyn}$) caused by the movement resulting the actual contact force as equation below.

$$F_c = F_d + \left(k_e \cdot \delta_{dyn}\right) = k_e(\delta_{sp} + \delta_{dyn}) \tag{8-4}$$

where δ_{dyn} is displacement due to the environment movement.

One practical way to maintain desired contact force on the environment and to prevent any large reaction force F_R, is by controlling the change in velocity between the robot and ball. As described in Chapter 7, this study proposes a Neuro-Fuzzy learning algorithm (NFLA) to compensate the main Neuro-Fuzzy force controller in order to maintain a low contact force whilst simultaneously tracking the environment movement by reducing the reaction force caused by the moving object. Furthermore, the problem of nonlinearities and unknown environment that introduces instability to the system can be solved. Considering the case of hard stiffness environment that moves towards the Stäubli robot along z axis at a low constant velocity, this will initially increase the rate of force error. Subsequently, its triggers the Neuro-Fuzzy controller to calculate an appropriate incremental position to minimize the force error while simultaneously tracking the environment movement. The following section will discuss the effectiveness of the proposed Neuro-Fuzzy Learning and Adaption (NFLA) mechanism on moving environment.

8.3.2 Test description

One of the key requirements of the system is to ensure sufficient fidelity of the proposed force controller to uncertainties and unknown contact environment as previously discussed. In order to demonstrate and analyze the controller performance, a series of tests were designed to evaluate the capability of proposed adaptive Neuro-Fuzzy force controller. Both conventional and intelligent methods were preliminary evaluated using several designed tasks which were intended to provide comparison results. The results were evaluated using performance measures as described in Chapter 7. In this section, the experiments were performed in three different conditions as follows;

- Condition 1 (C1) – represents a task that demonstrates the robot system's ability to compensate and track environment movement whilst maintaining a desired contact force.
- Condition 2 (C2) – requires the robot to compensate the disturbance causing by environment movement at different velocity profiles.
- Condition 3 (C3) – involves a situation in which the robot may perform a task on different environment stiffness with varying environment's velocity profiles.

On initialization of the system, the robot is commanded to move to an initial start position (Figure 8-8 (a)) in which the controller is set up with the *z* axis in force control mode with other 5 axes locked in position control. The robot is then moved downward along *z* axis at an approximate speed of 20mm/s until it makes contact with the environment (Figure 8-8 (b)). Once contact is made, the force controller is activated to maintain a 5N contact force. After approximately one second the PUMA robot is commanded to move using polynomial trajectory with maximum velocity of 1 mm/s. Each condition test was repeated with 3 different environment average velocity profiles of 4 mm/s, 20 mm/s and 40 mm/s. The sequence of tests was also repeated with low environment stiffness contact by reducing the ball air pressure. Throughout the tests, the *z* axis force and position was recorded and the results will be discussed in the following section.

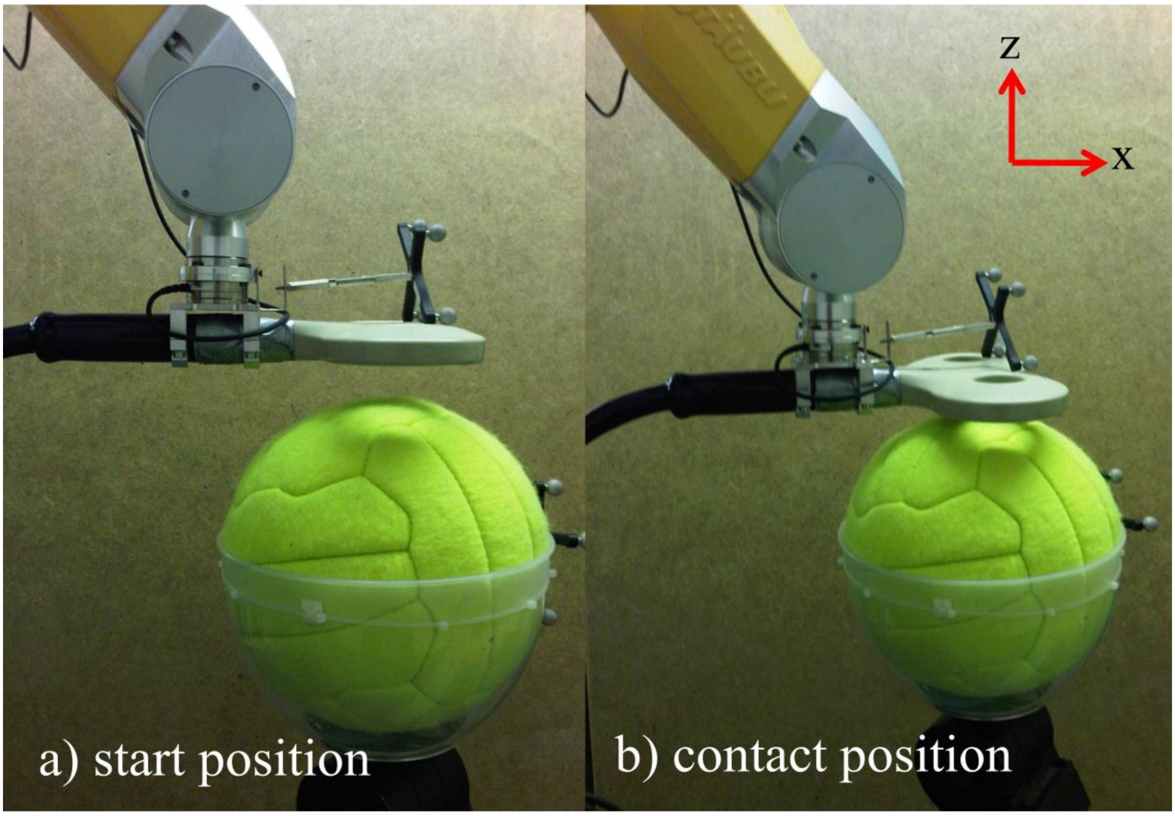


Figure 8-8 Stäubli and PUMA robot configuration initial position set up

8.3.3 Test results and discussion

8.3.3.1 *Condition 1 (Implementation of Neuro-Fuzzy learning and adaption (NFLA) mechanism)*

This test demonstrates the ability of the robot to track the environment movement while maintaining desired contact force, using Neuro-Fuzzy learning and adaption (NFLA) mechanism. The PUMA 560 robot is commanded to move using a polynomial trajectory with a maximum velocity of 4 mm/s. During each run, the PUMA 560 robot is required to travel towards the coil direction for a distance of 20 mm in a time of 5 seconds. Thereafter, the robot moves back to the starting position along the same axis as illustrated in Figure 8-9 (a). The results of Figure 8-9 provide the initial attempt of a main Neuro-Fuzzy force controller on moving environment without NFLA mechanism. It is observed that the Stäubli robot tracks the PUMA 560 robot movement with a maximum position error of 2.2 mm which is marked as ‘X’ in Figure 8-9 (a). This result corresponds to force response (Figure 8-9(b)), in which it is apparent that the maximum steady-state force error is approximately 4N during downward movement. This large steady state error in force response is due to the low value of the displacement to maintain a desired contact force δ_{sp} during downward movement. In this case the system senses the environment as soft. However, due to the nature of the control system, the resulting incremental positions appear to be insufficient to follow the PUMA 560 robot movement. This subsequently reduces the actual force response from the contact force demand. These errors were shown to be reduced and less significant at upward movement as the maximum force error is reduced to 1N. During upward movement the system predicts the environment as a higher stiffness environment in which the contact force was shown to be increased in direct proportion to an increase of δ_{sp}. It is also observable that the force response oscillates at high frequency of approximately 110 Hz for both movement directions, while the low frequency of 2.5 Hz is only seen in downward direction.

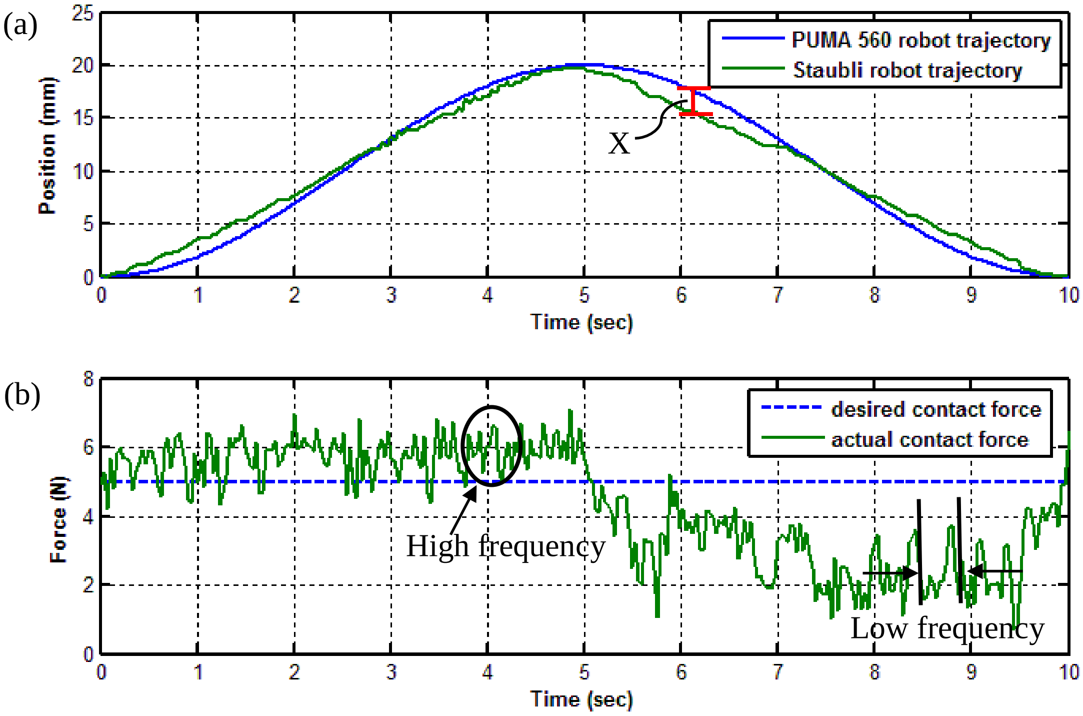


Figure 8-9 (a) PUMA 560 robot and Stäubli robot trajectories and (b) contact force response of Neuro-Fuzzy controller (without NFLA mechanism)

Figure 8-9 indicates that an increase in environment stiffness is accompanied by an increase in transient frequency, representing a higher force response bandwidth. This property made force control at low force level with low proportional gains particularly troublesome. The problem can be solved if higher values of controller gains were permitted when dealing with a low stiffness environment. Subsequently resulting in increasing speed of Stäubli robot to closely track the environment movement and subsequently reduce the force error.

In the interest of achieving a gradual convergence at the desired force setpoint for both conditions, an adjustment method was introduced to compensate the anticipated steady-state force error and unpredictable nature of response oscillatory behaviour. A similar test to that shown in Figure 8-9 was repeated and recorded in Figure 8-10. During this test, the NFLA mechanism is integrated to the system in order to enhance the force controller performance. The aim of this implementation is to reduce the large steady-state force error response, which is caused by environment movement as shown in Figure 8-9(b). Figure 8-10(b) shows the effectiveness of the proposed mechanism, in which it can be seen that the steady-state force error is extremely reduced. Clearly in the result of Figure 8-10(a), the Stäubli robot is closely tracking the PUMA 560 robot movement while successfully maintaining the desired contact force in both upward and downward directions. The other important observation is the reduced effect of uncontrollable oscillatory transient force response displayed in Figure 8-9(b) and virtually eradicated in Figure 8-10(b). As the rate of force error can be considered relatively small for upward movement, the system applied a small adjustment to speed up the Stäubli robot movement, indicating an increase in force response bandwidth to cope with the environment movement. A similar behaviour can be observed during downwards movement, as the mechanism performs an adequate adjustment to reduce the large steady state force error.

The results of Figure 8-10 agree with the sequence of simulation results in Chapter 5, which shows the improved position disturbance rejection is certainly an important advantage to be gained from high force controller gains. The performance index of IAE described in Chapter 7 was used again to evaluate the force response error of the system with NFLA mechanism implementation. The comparison of IAE performance index demonstrates the effectiveness of the force tracking can be achieved with the NFLA mechanism. The IAE index values for the system without NFLA mechanism are 1752.6 and 4326.8 for upward and downward movement respectively. Whereas, the IAE index values for NFLA mechanism are 68.99 and 70.37 for upward and downward movement respectively, indicating approximately 25 times improvement compared to the system without NFLA mechanism.

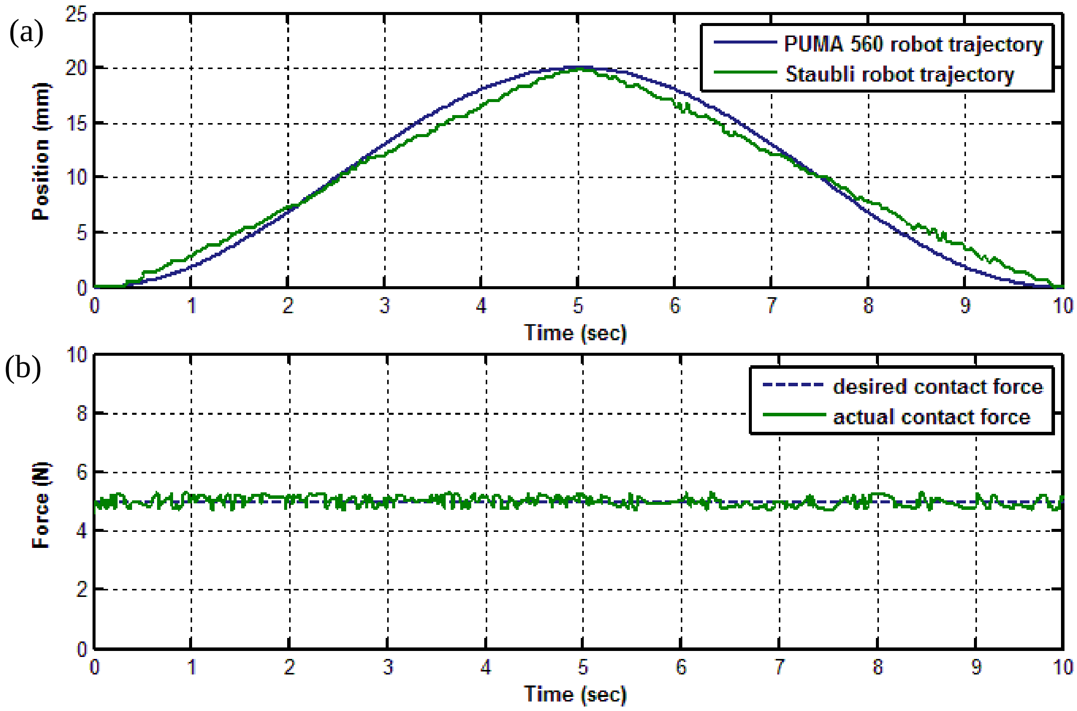


Figure 8-10 (a) PUMA 560 robot and Stäubli robot trajectories and (b) contact force response of NFLA mechanism

8.3.3.2 *Condition 2 (Environment velocity varying effects)*

Following the commissioning of NFLA mechanism, the previous comparison tests were undertaken to demonstrate the ability of the system, using the NFLA mechanism. It was shown that the NFLA mechanism appears as a reliable compensator of the environment movement, thus enhancing the system performance to maintain a desired contact force, while simultaneously tracking the environment movement. Having established this fact, a series of similar tests were repeated with varying environment velocity profiles to demonstrate the effect of varying environment velocity profile on the proposed force controller as in section 8.3.3.1. This test attempt to indicate that the environment velocity should has a significant effect to the system.

A similar test to that shown in section 8.3.3.1 was repeated with varying PUMA 560 robot maximum velocity. Identical force controller was used in these tests. Figure 8-11 provides the position and force responses when the PUMA 560 is commanded to move at maximum velocity of 20mm/s. Under this condition the results show an increase in environment velocity causes degradation in force response. However, Figure 8-11 shows a considerably improved force response to that shown in Figure 8-9(b) (force controller without NFLA at low velocity), less force error and less high frequency disturbance. The maximum observable steady-state errors in Figure 8-11(b) are approximately equal in both directions (~1N), indicating a consistent PUMA 560 robot maximum velocity that can be tolerated by the force controller. For an upward movement, it is observable that the force response profile tends to oscillate at high frequency while low frequency is observable in downward movement.

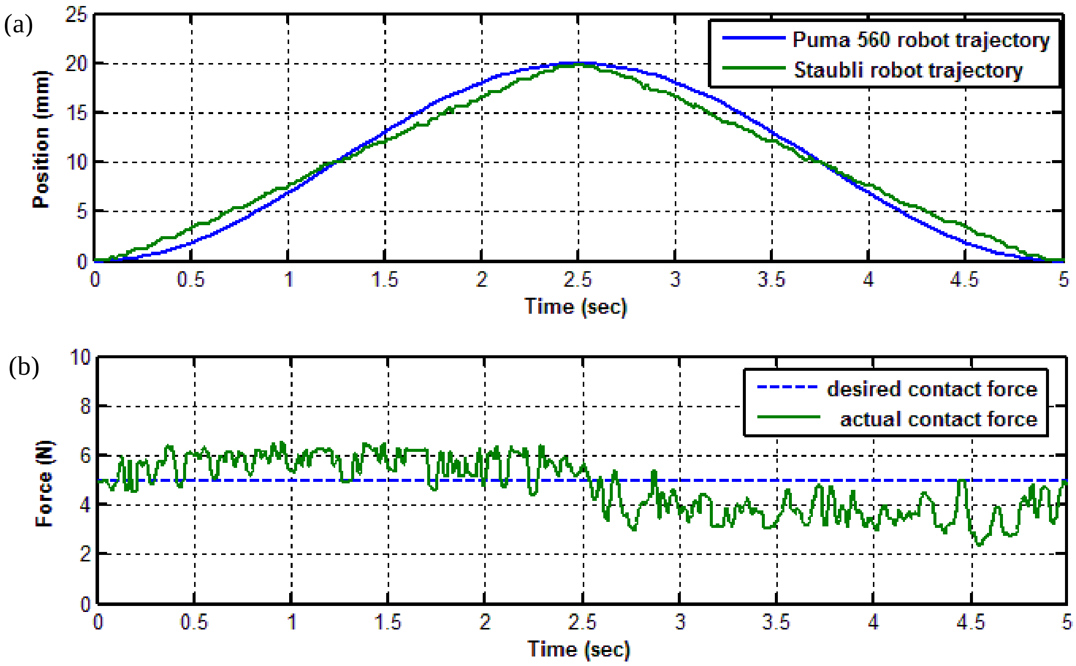


Figure 8-11 (a) PUMA 560 robot and Stäubli robot trajectories and (b) contact force response of NFLA mechanism with maximum environment velocity of 20mm/s

In principle, increasing PUMA 560 robot speed generates increased deflections of environment in upward direction (high stiffness) and reduced deflections of environment in downward movement direction (low stiffness). This effect results in consequential reduction in the net force within the control system. To confirm the significant of the environment varying maximum velocity, the test was repeated by increasing the PUMA 560 robot velocity further to v_e = 40 mm/s. The maximum force steady-state error in Figure 8-12(b) is approximately 15% higher than the steady-state error shown in Figure 8-11(b), while the PUMA 560 robot velocity used in Figure 8-12 is twice that used in Figure 8-11.

In a direct comparison between Figure 8-10 and Figure 8-12, the effect of v_e on force response is seen to be dramatic. This suggests that increasing the PUMA 560 robot velocity causes an increase in oscillatory nature of the force response in both movement directions. In Figure 8-12(b), for an upward movement, the system response presents peak to valley oscillations approximately 1.5N while a fluctuation (~3N) is observable during downward movement. It is expected that by increasing the PUMA 560 robot velocity further will result in repetitive bouncing action which can inevitably lead the system into instability. Thus it was decided to conduct all the tests up to this maximum velocity as shown in Figure 8-12(a), as it was too severe to proceed to prevent damage to robot and environment. Although the polynomial shape of both desired and actual position curves are of little general importance, it is interesting to note how closely PUMA 560 robot is followed by Stäubli robot for all tests.

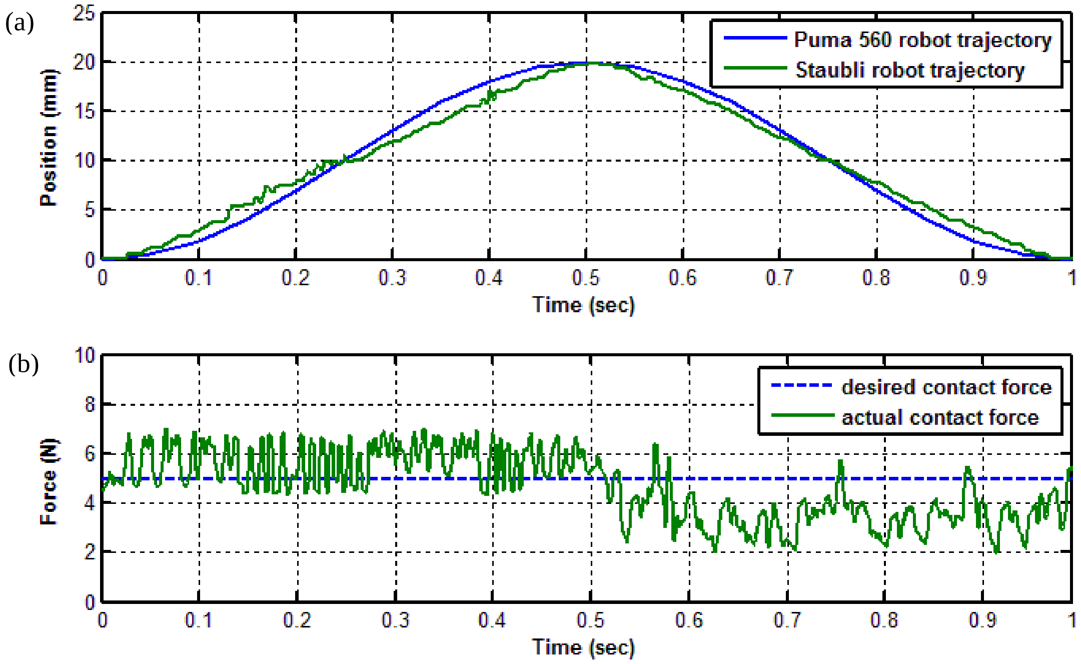


Figure 8-12 (a) PUMA 560 robot and Stäubli robot trajectories and (b) contact force response of with maximum environment velocity of 40 mm/s

This test was also used to determine the optimum velocity for the NFLA mechanism. However the attempt is slightly difficult to ascertain since it is largely qualitative in nature due to many uncontrollable and non-linear dynamics components of both robot system and the target environment such as backlash in the robot system as well as irregularity of surface environment. Therefore as the limitations of the dynamics components effects will not be consistent and unpredictable, it is proposed in this work to introduce general analogy that may be made to compensate these superimposed uncontrollable effects. That is controlled variation in working environment velocity can directly be compared to changes in sensed force. Thus, in selecting the optimum control parameters, a compromise had to be found between limiting the heavy and unpleasant applied force on subject's head, and avoiding robot arm ALTER limit axis speed (causing ALTER function to stop). These are also important safety features to prevent harm to the target environment causing from unwanted robot motions. It was concluded that the best performance in terms of system stability occurred for velocity in the region of ≤ 20mm/s and contact force of ≤ 10N, with the most significant improvements being in the maintaining a desired contact force.

8.3.3.3 *Condition 3 (Environment stiffness varying effects)*

The above experimental set up and procedure was repeated again in which other configuration parameters were held as near constant as possible, the exception being the environment stiffness. This was to enable the system stability and quality of the force controller to be evaluated, by examining the force response and displacement of the Stäubli robot in contact with much lower environment stiffness.

In order to provide the required compliance, it was decided that a simple and effective way of implementing the test would be to reduce air pressure of the ball which is the target environment of this study. Prior to conducting the test, it was necessary to determine the 'stiffness' of the environment by measuring the change in force with respect to the robot displacement taken over 30 seconds at fixed PUMA robot configuration as shown in Figure 8-8. Thus the environment can be considered to be a soft environment with stiffness of 4.23N/mm.

The tests were carried out with the similar velocity profiles as described in previous section 8.3.3.2. Figure 8-13 provides the result when the PUMA 560 robot at a lowest test's maximum velocity of 4mm/s. It is apparent from the graph that the system can maintain the desired contact force with observable force error of approximately 1N and 1.5N during upward and downward movement respectively. This was expected as the previous tests suggest that force response is affected if the contact environment is relatively low. The results of Figure 8.13 allow easy comparison with variation of environment stiffness. The results indicate that the force error is inversely proportionate to the environment stiffness. However, these errors are significantly less than those without NFLA mechanism. The similar behaviour with previous test in section 8.3.3.2 is observed. If the PUMA 560 robot moves upward, the system tended to oscillate at high frequency whilst low frequency is apparent in downward movement.

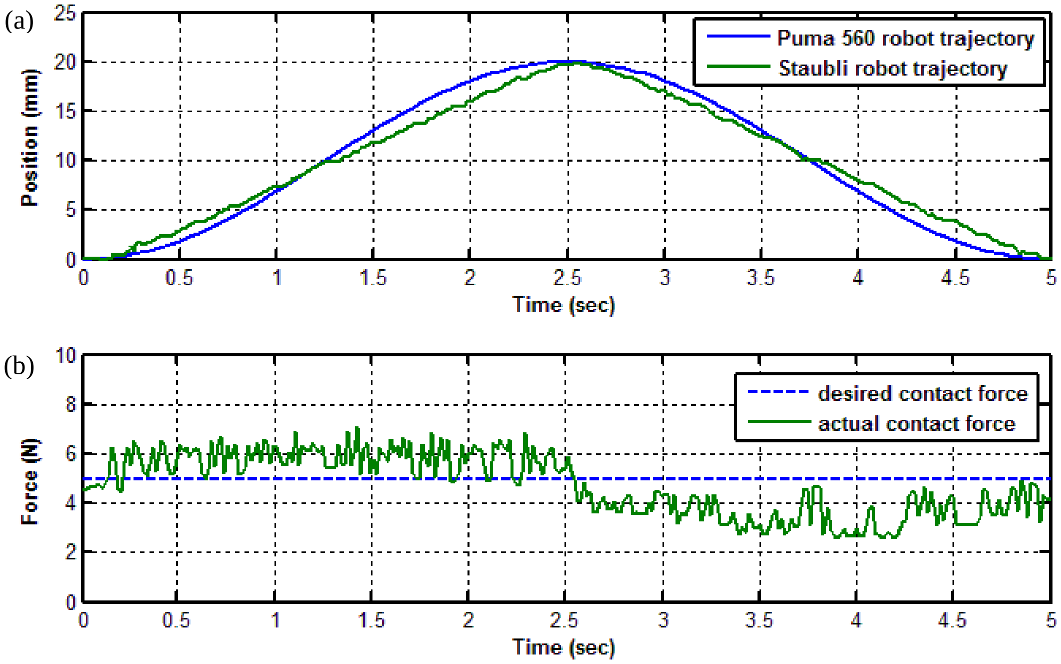


Figure 8-13 (a) PUMA 560 robot and Stäubli robot trajectories and (b) contact force response of NFLA mechanism with maximum environment velocity of 4 mm/s

Increasing the PUMA 560 robot velocity further to 20 mm/s in Figure 8-15, shows a clear trend of increasing oscillatory behaviour of the force response. During upward movement the force response shows a similar profile to Figure 8-13(b), where the force error values is approximately 1N. However, for a downward movement the system response presents a maximum peak to valley oscillations approximately 3.8N. This is mainly due to a downward movement of the PUMA 560 robot causing a decrease in the value of δ_{sp}. Subsequently, this results in an increase of robot speed to maintain a desired contact force while tracking the environment movement. As the PUMA 560 robot is always moving, an increase of the Stäubli robot speed contributes to an increase in oscillatory behaviour as shown in Figure 8-15(b).

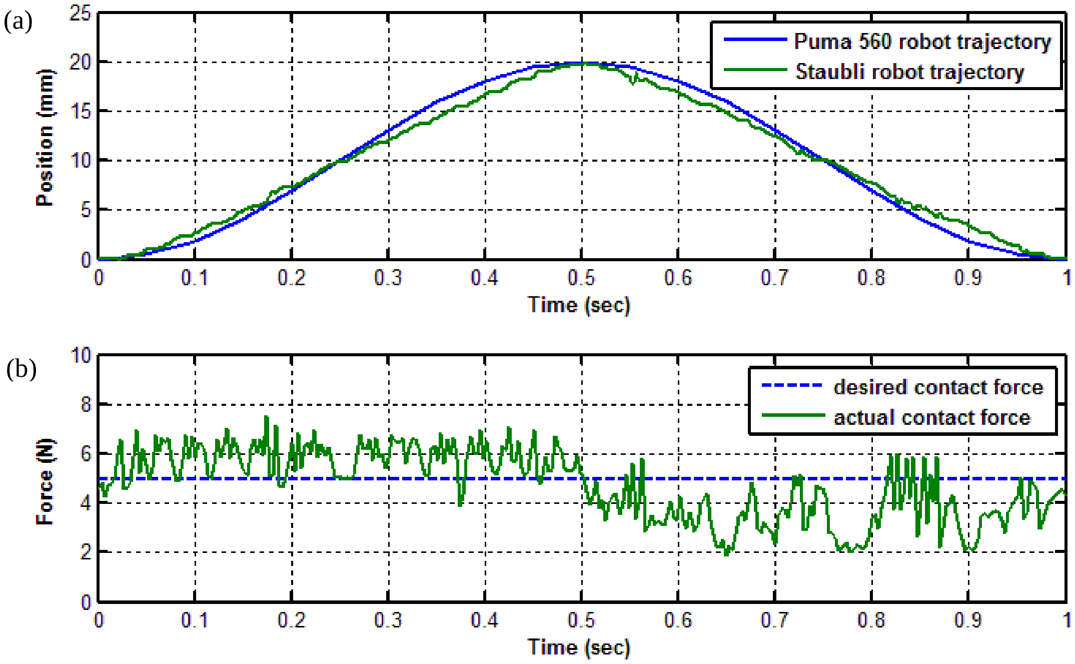


Figure 8-14 (a) PUMA 560 robot and Stäubli robot trajectories and (b) contact force response of NFLA mechanism with maximum environment velocity of 20 mm/s

Increasing the PUMA 560 robot velocity still further, provides the results of Figure 8-15, where v_e = 40mm/s. The force response in Figure 8-15(b) shows similar behaviour to those tests in Figure 8-13(b) and Figure 8-14(b). However, the effect of increasing environment velocity is shown to increase the oscillatory nature of the response and to cause a near instability condition during downward movement. In order to maintain a desired contact force it is preferable for the Stäubli robot to move approximately at a velocity equal to the environment. However, as the environment velocity is always varied, this sometimes result in exceeding desired contact force setpoint and subsequently causes the Stäubli robot to spring backward rapidly as marked as ‘X’ in Figure 8-15(a). This result corresponds to a large peak to volley oscillations of approximately 4.2N.

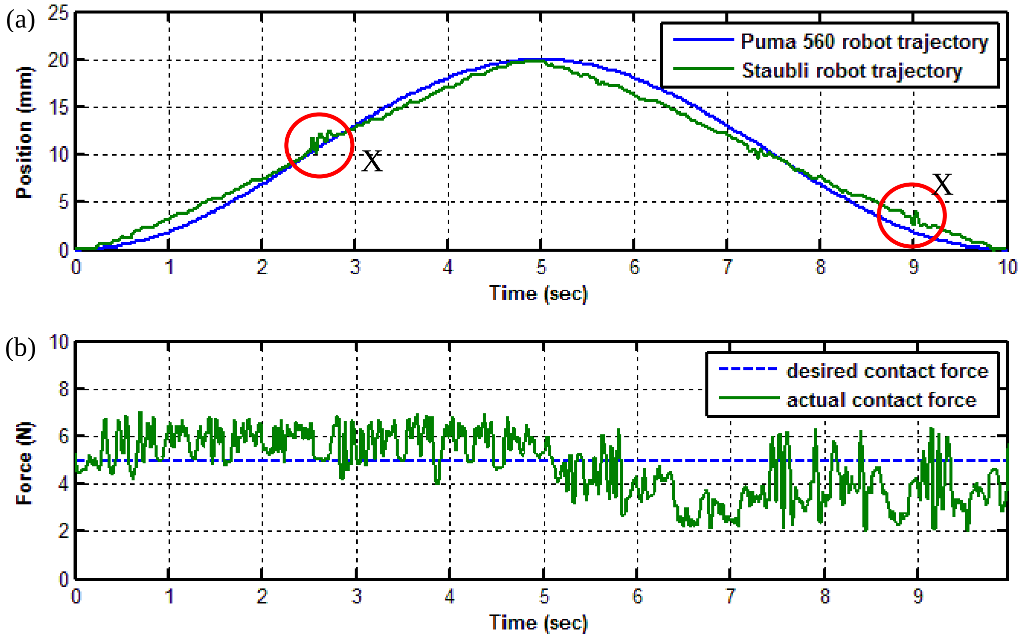


Figure 8-15 (a) PUMA 560 robot and Stäubli robot trajectories and (b) contact force response of NFLA mechanism with maximum environment velocity of 40mm/s

The qualitative IAE index performance of the system under different environment velocity profiles are given in following Figure 8-16. It is apparent from the graph that an increase in maximum velocity profile is accompanied by an increase in IAE index value in all conditions. Results obtained for the test of environment stiffness of 10N/mm show that the most significant difference in performance for maximum velocity profiles ranges from 4mm/s to 40mm/s. Whilst the flat portion of the curve of k_e = 4.23 N/mm illustrates there was no significant difference in performance in all conditions. In a direct comparison between Figure 8-16(a) and Figure 8-16(b), the movement direction on force response is seen to be dramatic. In both figures, for upward movement the IAE values are relatively lower than downward movement. Reduction in force error values were expected as discussed in section 8.3.3.2 owing to the much higher environment stiffness. The system changes to a more responsive system as a lower Stäubli robot velocity induces a much higher compensation of reaction force in the less compliance environment.

Comparing the results in Figure 8-16(a) and Figure 8-16(b), the highest IAE value for both environment stiffness were recorded at a similar maximum velocity during downward movement. This was expected, as it was found relatively difficult to maintain a low contact force during downward movement as the value of δ_{sp} could be easily reduced. As the environment was moving in this direction, the contact force error increases very rapidly. Consequently, a high value of the Stäubli robot speed is set by the controller. This result in exceeding desired contact force setpoint and subsequently causes the Stäubli robot to spring backward rapidly as can be seen in Figure 8-15. Increasing the environment maximum velocity further, might result in loss of contact with the environment and subsequently initiates inevitably limit cycle that creates a repetitive oscillating behaviour. To prevent this instability, the Stäubli robot maximum velocity is kept constant at 40mm/s to avoid any harm to the robot and environment.

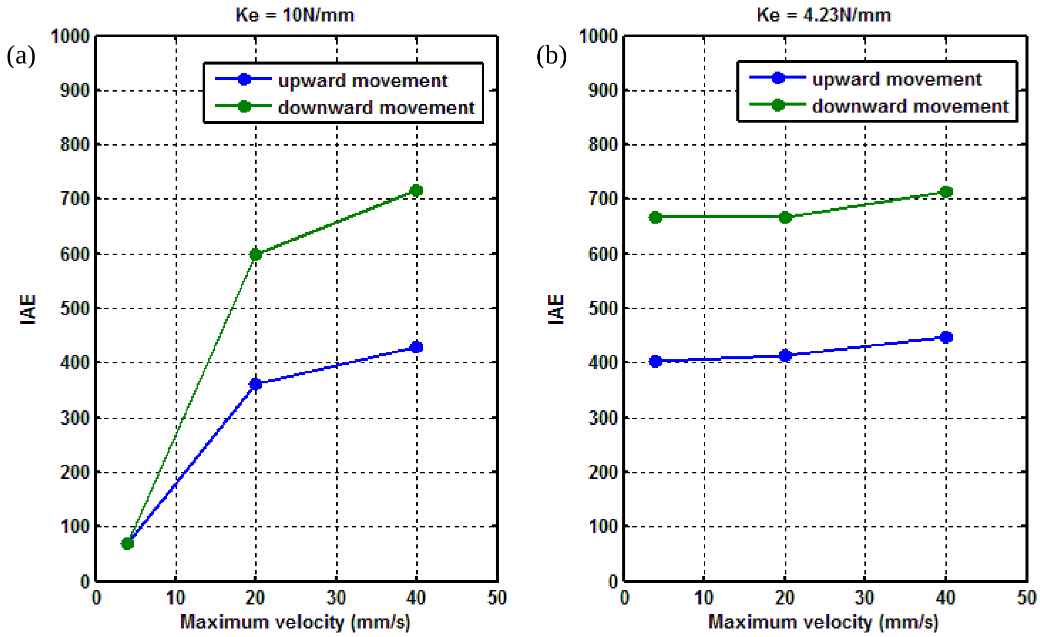


Figure 8-16 IAE performances indices

Overall, the force response is greatly influenced by the environment movement velocity and its direction. Due to the nature of an incremental position control, the control system stability has been shown to be difficult to achieve experimentally during downward movement. Combining both environment movement direction and the nature of incremental position control, the NFLA mechanism would give the best performance during environment's upward movement at maximum velocity of ≤ 40mm/s. Despite an increase value of IAE in Figure 8-16, the other important observation is the significant reduction of IAE value to 25 times lower as NFLA mechanism is implemented. Details were discussed in section 8.3.3.1. These results display an important benefit of NFLA mechanism in helping to overcome the problem of the system inability to respond to disturbances caused by moving environment.

8.4 Summary

Using the proposed force-controlled TMS robotic system in Chapter 4, an experimental study have been carried out, to investigate and demonstrate the ability of the proposed system in both TMS procedure namely preoperative and intraoperative phases. The circular tracking test (section 8.2.1) demonstrated that the robot arm movement can be directly controlled by the operator. The position error can be effectively controlled within less than 0.8mm with an applied force of approximately 3N.

The implementation of Neuro-Fuzzy force controller enabled the system force control gains to be optimized to achieve stable operation on unknown contact environment stiffness. The experimental results also show that Neuro-Fuzzy learning and adaption (NFLA) mechanism is necessary in the external force control loop in order to achieve good environment movement disturbance invariance. The potential of the adaptive Neuro-Fuzzy force controller has been further illustrated by performing several tests on different designed scenario tasks. Based on this, an investigation whether the proposed system can satisfy TMS procedure requirements could prove to be very beneficial.

CHAPTER 9

CONCLUSION AND RECOMMENDATIONS FOR FUTURE WORK

A novel force-controlled TMS robotic system has been successfully developed during this study. The 6 DOF TMS robotic system is able to maintain a desired contact force on unknown/varies environment stiffness using an intelligent decision making approach. A series of experimental work has been extensively carried out to validate the capability of the proposed system.

9.1 Conclusion

The thesis has presented a new development of the force-controlled TMS robotic system which allows an automated TMS procedure. The motivation behind this study was to assist neurologists in medical and clinical practice as well as to make stimulation of the cerebral cortex more reliable. Apart from providing an indication of TMS system performance, this new robotic system has its emphasis on real-time control, programmability and repeatability which the robot provides to the advantage for TMS.

The TMS robotic system architecture is based on two computers that communicate in a synchronized manner to perform the complete control functions. A QNX host PC was used to control the system in real-time and is dedicated to acquire data from the F/T DAQ system and then subsequently generates the robot desired incremental position through force control law. This passes the instruction down to the robot controller which performs the real-time path control and interface communications with the manipulator arm.

A variant of the external hybrid force control scheme has been successfully implemented. The main advantage of controlling force using an outer force control loop, whilst still retaining an inner position loop, is it’s relatively easy to implement since it does not require much modification to the existing robot control system. Although the force control scheme does not directly control force as in a torque based system, this approach offers reasonably good force control and reduces the system computational burdens. Furthermore, with force being controlled by modifying the robot position demand, the scheme allows simultaneous control of position and force control.

The problems associated with the stability of force control have been highlighted when initial attempt is made to carry out force control task. To establish the reason behind this, a single axis dynamic model of the TMS robotic system was developed and the simulation study was carried out to identify the factor affecting the control system stability. A good agreement is achieved between actual response tests and simulation. Several unconstrained and constrained tasks have been carried out to evaluate the overall performance of the system and the study has provided a firm foundation to develop an alternative force control law.

A well tuned PI controller has gain parameters which are adapted to the specific environment stiffness in which the system can accomplish desired response and satisfactory stable performance. However, if the environment stiffness varies without re-tuning the controller, the simulation and experimental results showed that this condition leads to a degraded performance and as a consequence reduced the stability in the control system. To overcome the stability problem associated with unknown/varies environment stiffness, an adaptive Neuro-fuzzy force controller has been developed to automatically adjust the control gains so that the stable force control is achieved.

In addition, the proposed method also employs a learning and adaptive mechanism to adjust a main loop Neuro-fuzzy force controller online. The purpose of this mechanism is to learn the environment and adapt the controller accordingly. Adaption consists of comparing the model reference error and the derivative of error as the input to the Neuro-fuzzy algorithm. Any undesired response is immediately compensated with a proper adjustment parameter according to the response behaviour. Experimental results have shown the effectiveness of the proposed controller particularly for nonlinearities as well as when the environment parameters and position are unknown and change substantially.

Note that, the response of the system when in contact with the moving environment has highlighted the rapid changes in force transients, which suggested that an increasing in environment maximum velocity is accompanied with the increasing in steady-state force error. In order to obtain an optimum velocity that the system can react, the compromise has been found between the heavy and unpleasant applied contact force and a maximum incremental position that the ALTER real-time path modification function can achieved. It was found during the experiment the current developed TMS robotic system can react with environment velocity less than 40mm/s.

A summary of achievements as follows;

- This study represents, for the first time, a detailed investigation and implementation of force control strategy on TMS robotic system with unknown knowledge of the subject’s head characteristics, and thereby constitutes an important contribution to the evaluation of the TMS robotic system.
- Designing and developing of a TMS robotic system which comprises hardware configuration, real-time software design and operational procedure of the TMS robotic system.
- Development of an Adaptive Neuro-Fuzzy Force Controller (ANNFC) to maintain a desired low contact force on unknown/varies environment stiffness whilst simultaneously tracking the environment movement.
- A series of experimental work has been extensively carried out to validate the capability of the designed and developed force-controlled TMS robotic system as mentioned above. The results show that the system is able to fulfil the TMS procedure requirement with robust and stable performance.

9.2 Recommendations for Future Work

The robotic system has potential for further work in order to fully establish the feasibility of the TMS robotic system. The use of CS8C robot controller as the real-time robot position controller in the present system means that the potential for introducing additional control features and advanced control scheme in the future is severely limited. It has been shown that complex and interrupt-driven software has had to be developed to enable the CS8C robot controller to drive the ALTER path modification, whilst it continues to acquire real-time data from the QNX host PC and perform all of the necessary calculations. It would be beneficial to enhance the current system by developing an independent open architecture controller to control the manipulator arm directly by the host PC carrying out the necessary kinematics and dynamics transformations.

The problem of present low sampling rate of Force/Torque (F/T) DAQ system has prevented the F/T sensor from being used to its full potential. The sampling rate and system bandwidth, which is fundamental to the TMS robotic system, has already been discussed. In order to achieve a stable and high fidelity force feedback control, it is important that improvements are made. An alternative method of acquiring the F/T data using the Network Force/Torque (NET F/T) sensor system interface should help to acquire the sensor data at much higher sampling rates than the present 500Hz. This realistic, possibility and a faster output rate of F/T sensor system interface has been developed by ATI Company. The system interface can measure up to 7000Hz for all six axis sensor measurement through Ethernet UDP protocol. However, the implementation of the NET F/T system in the current robotic system is limited, since the CS8C robot controller does not support the UDP protocol. If the NET F/T system can be used in conjunction with the open architecture robot controller, this will be a major importance in future work of force feedback control, when higher sampling rates will be vital for improved system performance.

Another option which has been identified is the replacement of the QNX host PC and the CS8C robot controller with a reconfigurable embedded control and acquisition system called CompactRIO. This flexible system is manufactured by National Instruments Corporation and was designed to offer the processing power for advanced control applications, high-speed data transfer and logging, and processing-intensive application such as advanced motion. The system also provides flexible NI LabVIEW graphical system design software to help the control system designer to develop a real-time application. Currently, there are more than 50 C Series modules for different measurements to accommodate a wide range of system requirements. For instance, the NI 7350 Series which integrate with NI DAQ devices can be used to create a powerful custom motion controller to drive the robot arm. Introducing a much more flexible system will provide the greatest promise for improved force control strategy.

Experimental work indicates significant improvements in the degree of stability are possible by implementing an intelligent decision-making control law to solve the uncertainties and variations in force control strategy. Since present adaptive Neuro-Fuzzy force controller inputs are fully depended on force error and rate of error, it is possible to integrate position/velocity information to enhance the controller decision making. The environment stiffness values can be predicted as the value is inversely proportional to the robot position.

It can be seen, therefore, that several realistic possibilities exists to improve the present robotic system, to ensure its continued operation in TMS application research. With the growing interest in robotic system in TMS application, this present system also has the potential for successful and long term involvement in research and development of medical system devices.

REFERENCES

Aström, K. J. (2002) *Control System Design* Available at: http://www.cds.caltech.edu/~murray/courses/cds101/fa02/caltech/astrom-ch6.pdf (Accessed: 10th Jan 2011).

Åström, K. J. and Wittenmark, B. (1995) *Adaptive control.* Addison-Wesley.

ATI. (2010) 'F/T Data Acquisition (DAQ)', *Electronic Hardware*, pp. B-17. [Online]. Available at: (Accessed:

Barker, A. T. and Jalinous, R. (1985) 'Non-Invasive Magnetic Stimulation of Human Motor Cortex', *Lancet,* 1, (8437), pp. 1106-1107.

Bigras, P., Lambert, M. and Perron, C. (2007) 'New formulation for an industrial robot force controller: Real-time implementation on a KUKA robot', *IEEE International Conference on Systems, Man and Cybernetics.* 7-10 Oct. 2007. pp. 2794-2799.

Bolton, W. (2002) *Control systems.* Newnes.

Burger, T., Laible, U. and Pritschow, G. (2001) 'Design and test of a safe numerical control for robotic surgery', *Cirp Annals-Manufacturing Technology,* 50, (1), pp. 295-298.

Burn, K. (1993) *Control of a force reflecting telerobotics system.* thesis. Newcastle University.

Burn, K., Short, M. and Bicker, R. (2003) 'Adaptive and nonlinear fuzzy force control techniques applied to robots operating in uncertain environments', *Journal of Robotic Systems*, 20, (7), pp. 391-400.

Cantello, R. (2002) 'Application of transcranial magnetic stimulation in movement disorders', *Journal Clinical Neurophysiology*, 19, (4), pp. 272-293.

Cavanaugh, J. T., Goldvasser, D., McGibbon, C. A. and Krebs, D. E. (2005) 'Comparison of head- and body-velocity trajectories during locomotion among healthy and vestibulopathic subjects', *Journal of Rehabilitation Research and Development*, 42, (2), pp. 191-198.

CE, E. C. f. S.-. (1998-2009) 'Guidelines for Classification of Medical Devices', in Consulting, W. T.(ed), *How to classify Medical Devices (MD)?* London, UK: Wellkang® Group.

Chan, R., Rogers, D. K. and McCloskey, D. I. (1996) 'Postural stability of the head in response to slowly imposed, small elastic loads', *Neuroscience Letters,* 214, (2–3), pp. 205-207.

Davies, B. (2000) 'A review of robotics in surgery', *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine,* 214, (1), pp. 129-140.

De Schutter, J. (1987) 'A study of active compliant motion control methods for rigid manipulators based on a generic scheme*', Robotics and Automation. Proceedings. 1987 IEEE International Conference on.* Mar 1987. pp. 1060-1065.

De Schutter, J. and Van Brussel, H. (1988a) 'Compliant Robot Motion I. A Formalism for specifying compliant motion tasks', *International Journal of Robotics Research,* 7, (4), pp. 3-17.

De Schutter, J. and Van Brussel, H. (1988b) 'Compliant Robot Motion II. A Control Approach Based On External Control Loops', *International Journal of Robotics Research,* 7, (4), pp. 18-33.

Degoulange, E. and Dauchez, P. (1994) 'External force control of an industrial PUMA 560 robot', *Journal of Robotic Systems,* 11, (6), pp. 523-540.

Degoulange, E., Dauchez, P. and Pierrot, F. (1993a) 'Determination of a force control law for an industrial robot in contact with a rigid environment*', IEEE International Conference on Systems, Man and Cybernetics.* Le Touquet, Fr, pp. 270-275.

Degoulange, E., Dauchez, P. and Pierrot, P. (1993b) 'Force control of an industrial Puma 560 robot under environmental constraints: Implementation issues and experimental results*', IEEE International Conference on Robotics and Automation.* Atlanta, GA, USA, pp. 213-218.

Dhillon, B. S. and Fashandi, A. R. M. (1997) 'Safety and reliability assessment techniques in robotics', *Robotica,* 15, pp. 701-708.

Dombre, E., Duchemin, G., Poignet, P. and Pierrot, F. (2003) 'Dermarob: A safe robot for reconstructive surgery', *IEEE Transactions on Robotics and Automation*, 19, (5), pp. 876-884.

Dombre, E. and Khalil, W. (2007) 'Robot Manipulators: Modeling, Performance Analysis and Control', in HAL - CCSD.

Dombre, E., Poignet, P., Pierrot, F., Duchemin, G. and Urbain, L. (2001) 'Intrinsically safe active robotic systems for medical applications ', *Proc 1st IARP/IEEE-RAS Joint Workshop on Technical Challenge for Dependable Robots in Human Environment* pp. 1-4.

Dorf, R. C. and Bishop, R. H. (2008) *Modern control systems.* Pearson Prentice Hall.

Driankov, D., Hellendoorn, H. and Reinfrank, M. (1996) *An introduction to fuzzy control.* Springer.

Duchemin, G., Poignet, P., Dombre, E. and Pierrot, F. (2004) 'Medically safe and sound', *IEEE Robotics & Automation Magazine*, 11, (2), pp. 46-55.

Duda, R. O., Avendano, C. and Algazi, V. R. (1999) 'An adaptable ellipsoidal head model for the interaural time difference', *IEEE International Conference onAcoustics, Speech, and Signal Processing, 1999.* . 15-19 Mar 1999. pp. 965-968 vol.2.

Eppinger, S. D. and Seering, W. P. (1987a) 'Introduction to Dynamic models for robot force control', *IEEE Control Systems Magazine*, 7, (2), pp. 48-52.

Eppinger, S. D. and Seering, W. P. (1987b) ' Understanding Bandwidth Limitations in Robot Force Control', *IEEE International Conference on Robotics and Automation.* Raleigh, NC, USA, IEEE, pp. 904-909.

Ettinger, G. J., Leventon, M. E., Grimson, W. E., Kikinis, R., Gugino, L., Cote, W., Sprung, L., Aglio, L., Shenton, M. E., Potts, G., Hernandez, V. L. and Alexander, E. (1998) 'Experimentation with a transcranial magnetic stimulation system for functional brain mapping', *Med Image Anal*, 2, (2), pp. 133-42.

Ferman, L., Collewijn, H., Jansen, T. C. and Van den Berg, A. V. (1987) 'Human gaze stability in the horizontal, vertical and torsional direction during voluntary head movements, evaluated with a three-dimensional scleral induction coil technique', *Vision Research*, 27, (5), pp. 811-828.

Finke, M., Fadini, T., Kantelhardt, S., Giese, A., Matthaus, L. and Schweikard, A. (2008) 'Brain-Mapping using robotized TMS', *International Conference of the IEEE Engineering in Medicine and Biology Society.* Vancouver, CANADA, Aug 20-24. pp. 3929-3932.

Flash, T. and Hogan, N. (1985) 'The coordination of arm movements: an experimentally confirmed mathematical model', *The journal of Neuroscience,* 5, (7), pp. 1688-1703.

Gadoue, S. (2007) 'Industrial Automation - Artificial Neural Networks', [Online]. Available at: (Accessed:

Gambier, A. (2004) 'Real-time control systems: a tutorial', *5th Asian Control Conference, 2004.* 20-23 July 2004. pp. 1024-1031 Vol.2.

George, M. S., Lisanby, S. H. and Sackeim, H. A. (1999) 'Transcranial magnetic stimulation - Applications in neuropsychiatry', *Archives of General Psychiatry,* 56, pp. 300-311.

Gresty, M. (1987) 'Stability of the head: studies in normal subjects and in patients with labyrinthine disease, head tremor, and dystonia', *Movement disorders,* 2, (3), pp. 165-185.

Hagemann, A., Rohr, K., Stiehl, H. S., Spetzger, U. and Gilsbach, J. M. (1999) 'Biomechanical modeling of the human head for physically based, nonrigid image registration', *Medical Imaging, IEEE Transactions on*, 18, (10), pp. 875-884.

Hallett, M. (2007) 'Transcranial magnetic stimulation: A primer', *Neuron,* 55, (2), pp. 187-199.

Hannaford, B., Kim, W. S., Lee, S. H. and Stark, L. (1986) 'Neurological control of head movements: Inverse modeling and electromyographic evidence', *Mathematical Biosciences,* 78, (2), pp. 159-178.

Haugen, F. (2004) *PID control.* Tapir Academic.

Herwig, U., Schonfeldt-Lecuona, C., Wunderlich, A. P., von Tiesenhausen, C., Thielscher, A., Walter, H. and Spitzer, M. (2001) 'The navigation of transcranial magnetic stimulation', *Psychiatry Research-Neuroimaging,* 108, (2), pp. 123-131.

Hogan, N. (1985a) 'Impedance Control: An Approach to Manipulation: Part I - Theory', *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME,* 107, (1), pp. 1-7.

Hogan, N. (1985b) 'Impedance Control: An Approach to Manipulation: Part II - Implementation', *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME,* 107, (1), pp. 8-16.

Hogan, N. (1988) 'On the stability of manipulators performing contact tasks', *IEEE Journal of Robotics and Automation,* 4, (6), pp. 677-686.

Ioannou, P. A. and Sun, J. (1996) *Robust adaptive control.* PTR Prentice-Hall.

Jang, J.-S. R. (1993) 'ANFIS: adaptive-network-based fuzzy inference system', *IEEE Transactions on Systems, Man and Cybernetics,* 23, (3), pp. 665-685.

Jang, J.-S. R. and Sun, C.-T. (1995) 'Neuro-fuzzy modeling and control', *Proceedings of the IEEE,* 83, (3), pp. 378-406.

Jang, J. S. R., Sun, C. T. and Mizutani, E. (1997) *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence.* Prentice Hall.

Jantzen, J. (2007) *Foundations of fuzzy control.* John Wiley & Sons.

Kazanzides, P., Fichtinger, G., Hager, G. D., Okamura, A. M., Whitcomb, L. L. and Taylor, R. H. (2008) 'Surgical and interventional robotics - Core concepts, technology, and design', *IEEE Robotics & Automation Magazine,* 15, (2), pp. 122-130.

Kazerooni, H. (1987) 'Robust, Non-linear impedance control for robot manipulators', *IEEE International Conference on Robotics and Automation.* Raleigh, NC, USA, IEEE, pp. 741-750.

Khatib, O. (1987) 'A Unified Approach for Motion and Force Control of Robot Manipulators - The Operational Space Formulation', *IEEE Journal of Robotics and Automation,* 3, (1), pp. 43-53.

Kiguchi, K. and Fukuda, T. (1997) 'Intelligent position/force controller for industrial robot manipulators - application of fuzzy neural networks', *IEEE Transactions on Industrial Electronics,* 44, (6), pp. 753-761.

Kiguchi, K. and Fukuda, T. (1999) 'Fuzzy selection of fuzzy-neuro robot force controllers in an unknown environment', *IEEE International Conference onRobotics and Automation, 1999.* . 1999. pp. 1182-1187 vol.2.

Korb, W., Engel, D., Boesecke, R., Eggers, G., Marmulla, R., O'Sullivan, N., Raczkowsky, J. and Hassfeld, S. (2003) 'Risk analysis for a reliable and safe surgical robot system', *Cars 2003: Computer Assisted Radiology and Surgery, Proceedings*, 1256, pp. 766-770.

Korb, W., Kornfeld, M., Birkfellner, W., Boesecke, R., Figl, M., Fuerst, M., Kettenbach, J., Vogler, A., Hassfeld, S. and Kornreif, G. (2005) 'Risk analysis and safety assessment in surgical robotics: A case study on a biopsy robot', *Minimally Invasive Therapy & Allied Technologies,* 14, (1), pp. 23-31.

Lancaster, J. L., Narayana, S., Wenzel, D., Luckemeyer, J., Roby, J. and Fox, P. (2004) 'Evaluation of an image-guided, robotically positioned transcranial magnetic stimulation system', *Human Brain Mapping,* 22, (4), pp. 329-340.

Lebosse, C., Renaud, P., Bayle, B., De Mathelin, M., Piccin, O. and Foucher, J. (2008) 'A robotic system for automated image-guided Transcranial Magnetic Stimulation', *IEEE/NIH Life Science Systems and Applications Workshop, LISA.* Bethesda, MD, pp. 55-58.

Li, H.-X. and Gatland, H. B. (1995) 'New methodology for designing a fuzzy logic controller', *IEEE Transactions on Systems, Man and Cybernetics,* 25, (3), pp. 505-512.

Li, H. X. and Gatland, H. B. (1996) 'Conventional fuzzy control and its enhancement', *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,* 26, (5), pp. 791-797.

Lin, S. T. and Huang, A. K. (1997) 'Position-Based Fuzzy Force Control for Dual Industrial Robots', *Journal of Intelligent and Robotic Systems: Theory and Applications,* 19, (4), pp. 393-409.

Lin, S. T. and Huang, A. K. (1998) 'Hierarchical fuzzy force control for industrial robots', *IEEE Transactions on Industrial Electronics,* 45, (4), pp. 646-653.

Lippmann, R. (1987) 'An introduction to computing with neural nets', *ASSP Magazine, IEEE,* 4, (2), pp. 4-22.

Mamdani, E. H. (1974) 'Application of Fuzzy Algorithms for Control of Simple Dynamic Plant', *Proceedings of the Institution of Electrical Engineers*, 121, (12), pp. 1585-1588.

Mason, M. T. (1981) 'Compliance and Force Control For Computer Controlled Manipulators', *IEEE Transactions on Systems, Man and Cybernetics*, SMC-11, (6), pp. 418-432.

MathWorks. (2011) 'Fuzzy Logic Toolbox', *Sugeno-Type Fuzzy Inference*, [Online]. Available at: http://www.mathworks.co.uk/help/toolbox/fuzzy/fp49243.html (Accessed: 01/05/2011).

Matthäus, L., Giese, A., Wertheimer, D. and Schweikard, A. (2006a) 'Planning and analyzing robotized TMS using virtual reality', *Studies in health technology and informatics*, 119, pp. 373-378.

Matthäus, L., Trillenberg, P., Bodensteiner, C., A., G. and A., S. (2006b) 'Robotized TMS for motion compensated navigated brain stimulation', *International Journal of Computer Assisted Radiology and Surgery*, 1, (SUPPL. 7), pp. 139-141.

Moore, S. T., Hirasaki, E., Raphan, T. and Cohen, B. (2005) 'Instantaneous rotation axes during active head movements', *Journal of Vestibular Research*, 15, (2), pp. 73-80.

Müller, B., Reinhardt, J. and Strickland, M. T. (1995) *Neural networks: an introduction.* Springer.

Ogata, K. (2009) *Modern control engineering.* Prentice Hall.

Ow, S. M. (1997) *Force Control of Teleroboticss.* thesis. University of Newcastle upon Tyne, UK.

Passino, K. M. and Yurkovich, S. (1998) *Fuzzy control.* Addison-Wesley.

Perdereau, V. and Drouin, M. (1993) 'A New Scheme for Hybrid Force-Position Control', *Robotica*, 11, pp. 453-464.

Peterson, B. W. and Richmond, F. J. (1988) *Control of head movement.* Oxford University Press, USA.

Pierrot, F., Dombre, E., Degoulange, E., Urbain, L., Caron, P., Boudet, S., Gariepy, J. and Megnien, J. L. (1999) 'Hippocrate: a safe robot arm for medical applications with force feedback', *Med Image Analysis*, 3, (3), pp. 285-300.

Pierrot, F., Dombre, E., Teot, L. and Degoulange, E. (2000) 'Robotized reconstructive surgery: ongoing study and first results', *IEEE International Conference on Robotics and Automation* pp. 1615-1620.

Plamondon, R. (1995) 'A kinematic theory of rapid human movements - Part I. Movement representation and generation', *Biological Cybernetics*, 72, (4), pp. 295-307.

Plamondon, R., Feng, C. and Woch, A. (2003) 'A kinematic theory of rapid human movement. Part IV: A formal mathematical proof and new insights', *Biological Cybernetics*, 89, (2), pp. 126-138.

Po-ngaen, W. (2006) *Neuro-Fuzzy Control in Tele-Robotics.* thesis. Newcastle University.

Poignet, P., Dombre, E., Merigeaux, O., Pierrot, F. and Duchemin, G. (2003) 'Design and control issues for intrinsically safe medical robots', *Industrial Robot-an International Journal*, 30, (1), pp. 83-88.

QNX (2004-2012) *QNX Neutrino RTOS*. Available at: (Accessed:
Raibert, M. H. and Craig, J. J. (1981) 'Hybrid Position-Force Control of Manipulators', *Journal of Dynamic Systems Measurement and Control-Transactions of the ASME*, 103, (2), pp. 126-133.

Reznik, L. (1997) *Fuzzy controllers.* Newnes.

Ross, T. J. (2004) *Fuzzy logic with engineering applications.* John Wiley.

Ruohonen, J. (2003) 'Chapter 1 Background physics for magnetic stimulation', in *Supplements to Clinical Neurophysiology*. Vol. 56 pp 3-12.

Salisbury, J. K. (1980) 'Active Stiffness Control of a Manipulator in Cartesian Coordinates', *Proceedings of the IEEE Conference on Decision and Control.* Albuquerque, NM, 10 December 1980 through 12 December 1980. IEEE, pp. 95-100.

Seraji, H., Lim, D. and Steele, R. (1996) 'Experiments in contact control', *Journal of Robotic Systems*, 13, (2), pp. 53-73.

Seul, J. and Hsia, T. C. (1998) 'Neural network impedance force control of robot manipulator', *Industrial Electronics, IEEE Transactions on*, 45, (3), pp. 451-461.

Short, M. and Burn, K. (2007) 'Robust and stable robotic force control', *4th International Conference on Informatics in Control, Automation and Robotics.* Angers, pp. 256-261.

Sommer, M., Tergau, F., Wischer, S. and Paulus, W. (2001) 'Paired-pulse repetitive transcranial magnetic stimulation of the human motor cortex', *Experimental brain research*, 139, (4), pp. 465-472.

Stankovic, J. A. (1988) 'Misconceptions about real-time computing: a serious problem for next-generation systems', *Computer*, 21, (10), pp. 10-19.

Stephen, P. (1996) *Bodyspace: anthropometry, ergonomics, and the design of work.* Taylor & Francis.

Visioli, A. (2006) *Practical PID control.* Springer Verlag.

Volpe, R. and Khosla, P. (1993) 'A theoretical and experimental investigation of explicit force control strategies for manipulators', *Automatic Control, IEEE Transactions on*, 38, (11), pp. 1634-1650.

Wasserman, P. D. (1989) *Neural computing: theory and practice.* Van Nostrand Reinhold.

Wassermann, E., Wassermann, E. M., Epstein, C. M. and Ziemann, U. (2008) *The Oxford handbook of transcranial stimulation.* Oxford University Press, USA.

Werbos, P. (1974) *Beyond regression: New tools for prediction and analysis in the behavioral science.* thesis. Harvard University.

Whitney, D. E. (1987) 'Historical-Perspective and State-of-the-Art in Robot Force Control', *International Journal of Robotics Research*, 6, (1), pp. 3-14.

Winters, J. (1988) 'Biomechanical Modeling of the Human Head and Neck', in F.J.Richmond, B. W. P. a.(ed), *Control of Head Movement.* Oxford University Press.

Yen, J. and Langari, R. (1999) *Fuzzy logic: intelligence, control, and information.* Prentice Hall.

Yi, X. (2012) *Design of a TMS Robotic System.* thesis. Newcastle University.

Yuen, S. G., Perrin, D. P., Vasilyev, N. V., del Nido, P. J. and Howe, R. D. (2010) 'Force tracking with feed-forward motion estimation for beating heart surgery', *Robotics, IEEE Transactions on*, 26, (5), pp. 888-896.

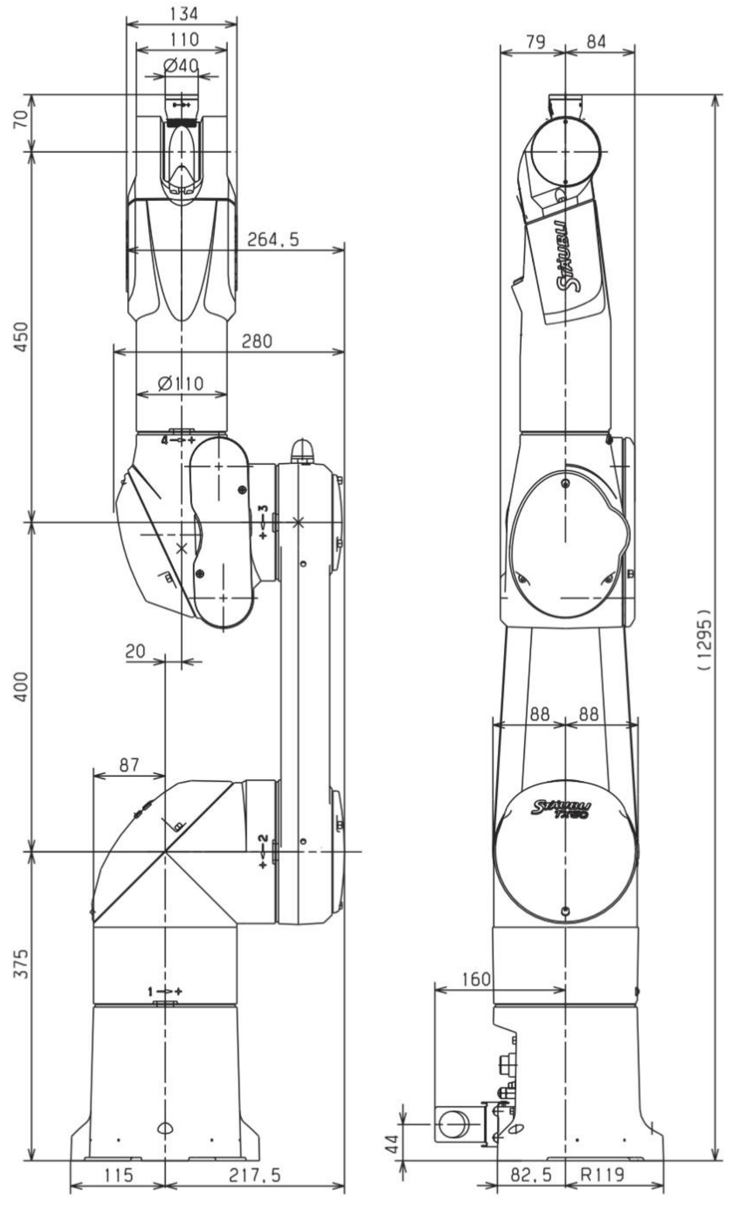
Zeng, G. W. and Hemami, A. (1997) 'An overview of robot force control', *Robotica*, 15, pp. 473-482.

Zhen-Yu, Z., Tomizuka, M. and Isaka, S. (1993) 'Fuzzy gain scheduling of PID controllers', *Systems, Man and Cybernetics, IEEE Transactions on*, 23, (5), pp. 1392-1398.

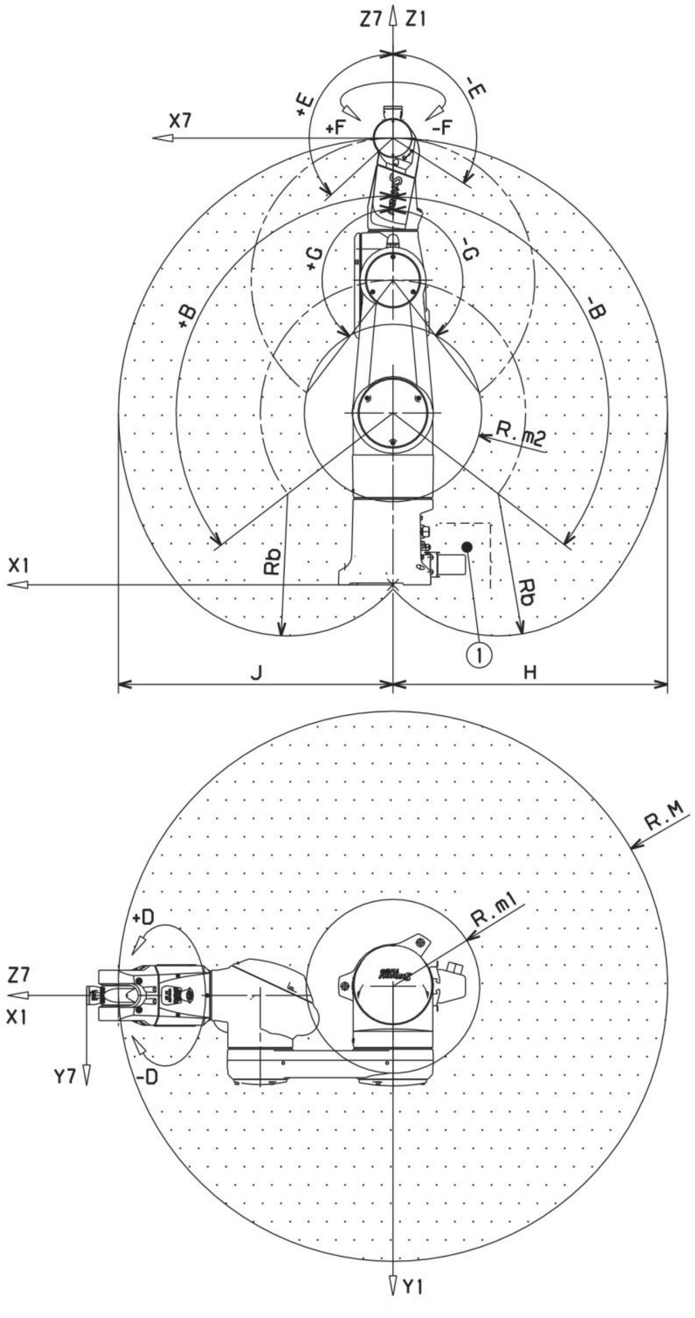
APPENDICES

APPENDIX A- 6 DOF ROBOT ARM SYSTEM

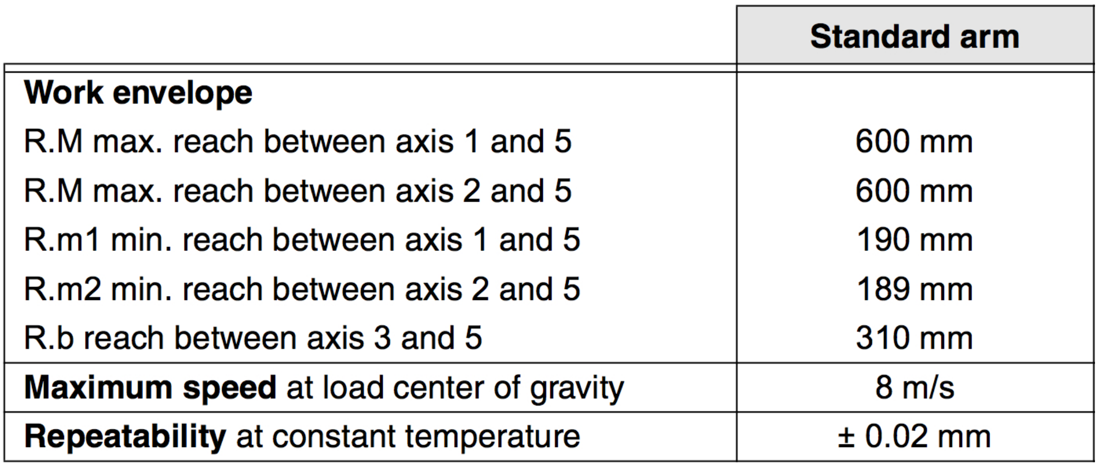
A-1 TX 60 robot arm dimensions



A-2 TX 60 robot arm workspace

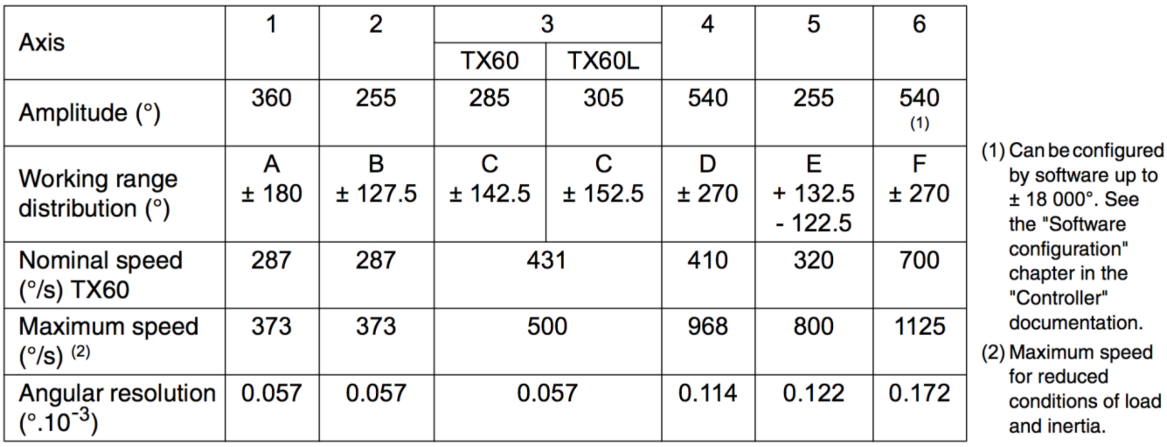


A-3 Robot arm performance



	Standard arm
Work envelope	
R.M max. reach between axis 1 and 5	600 mm
R.M max. reach between axis 2 and 5	600 mm
R.m1 min. reach between axis 1 and 5	190 mm
R.m2 min. reach between axis 2 and 5	189 mm
R.b reach between axis 3 and 5	310 mm
Maximum speed at load center of gravity	8 m/s
Repeatability at constant temperature	± 0.02 mm

A-4 Amplitude, speed and resolution

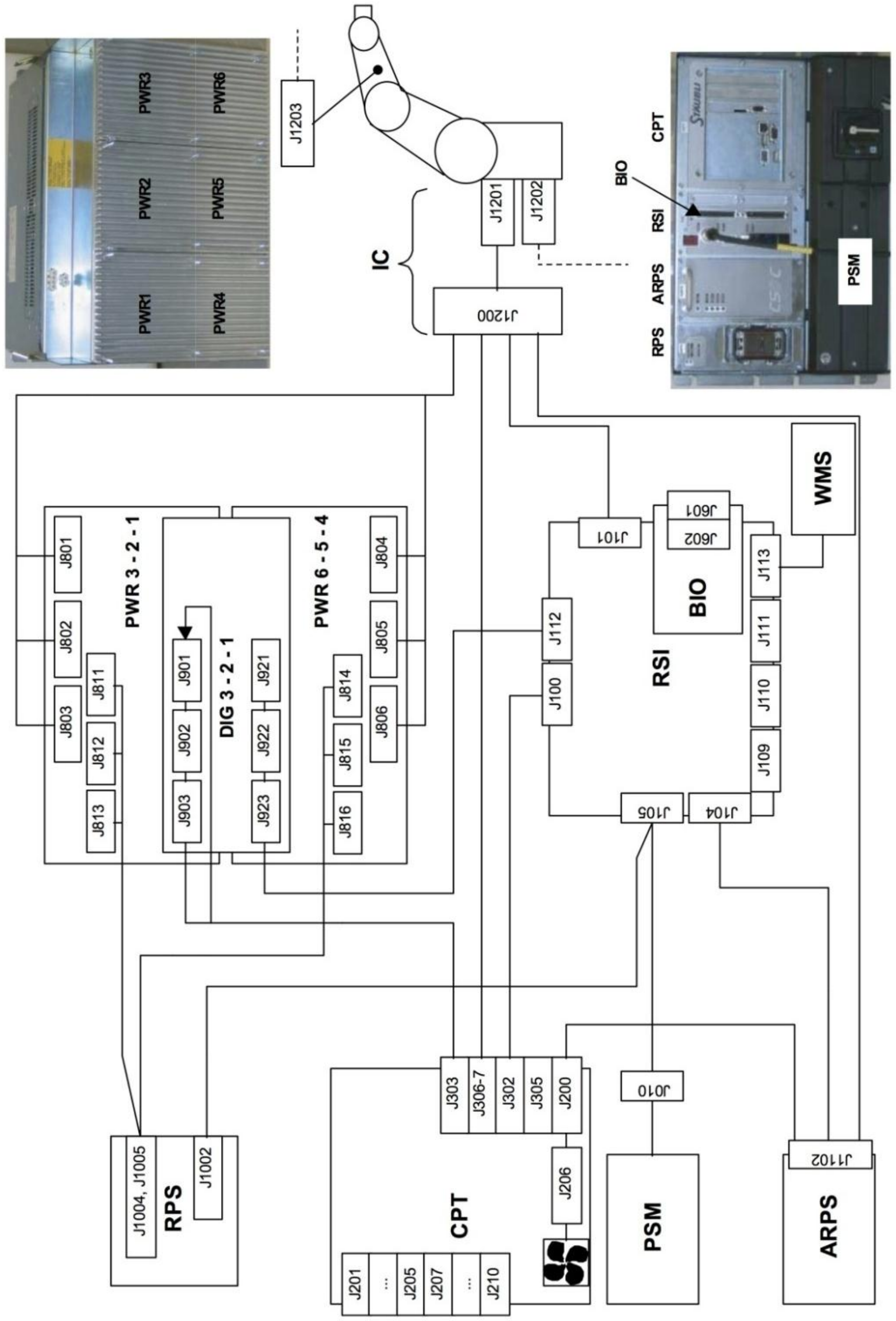


Axis	1	2	3		4	5	6
			TX60	TX60L			
Amplitude (°)	360	255	285	305	540	255	540 (1)
Working range distribution (°)	A ± 180	B ± 127.5	C ± 142.5	C ± 152.5	D ± 270	E + 132.5 - 122.5	F ± 270
Nominal speed (°/s) TX60	287	287	431		410	320	700
Maximum speed (°/s) (2)	373	373	500		968	800	1125
Angular resolution (°.10^{-3})	0.057	0.057	0.057		0.114	0.122	0.172

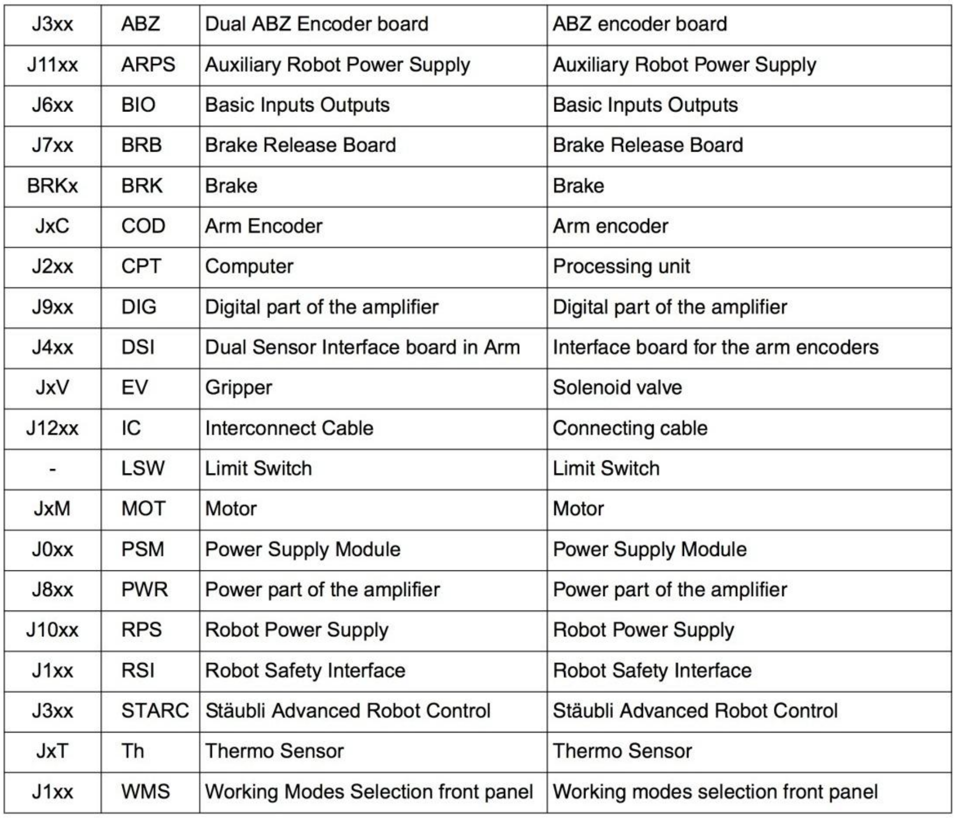
(1) Can be configured by software up to ± 18 000°. See the "Software configuration" chapter in the "Controller" documentation.

(2) Maximum speed for reduced conditions of load and inertia.

A-5 CS8C Robot Controller



A-6 CS8C robot controller glossary



J3xx	ABZ	Dual ABZ Encoder board	ABZ encoder board
J11xx	ARPS	Auxiliary Robot Power Supply	Auxiliary Robot Power Supply
J6xx	BIO	Basic Inputs Outputs	Basic Inputs Outputs
J7xx	BRB	Brake Release Board	Brake Release Board
BRKx	BRK	Brake	Brake
JxC	COD	Arm Encoder	Arm encoder
J2xx	CPT	Computer	Processing unit
J9xx	DIG	Digital part of the amplifier	Digital part of the amplifier
J4xx	DSI	Dual Sensor Interface board in Arm	Interface board for the arm encoders
JxV	EV	Gripper	Solenoid valve
J12xx	IC	Interconnect Cable	Connecting cable
-	LSW	Limit Switch	Limit Switch
JxM	MOT	Motor	Motor
J0xx	PSM	Power Supply Module	Power Supply Module
J8xx	PWR	Power part of the amplifier	Power part of the amplifier
J10xx	RPS	Robot Power Supply	Robot Power Supply
J1xx	RSI	Robot Safety Interface	Robot Safety Interface
J3xx	STARC	Stäubli Advanced Robot Control	Stäubli Advanced Robot Control
JxT	Th	Thermo Sensor	Thermo Sensor
J1xx	WMS	Working Modes Selection front panel	Working modes selection front panel

A-7 6 DOF Robot Arm System Comparison

Model Parameters		**Adept s650**	**Stäubli TX60**	**Denso VS6556G**	**Kawasaki FS003N**
Reach		653mm	670mm	650mm	620mm
Maximal Payload		5kg	9kg	6kg	3kg
Repeatability		±0.02mm	±0.02mm	±0.02mm	±0.05mm
Weight		28kg	51.4kg	35kg	20kg
Motion Range	Axis 1	±170°	±180°	±170°	±160°
	Axis 2	+45°,-190°	±127.5°	+135°,-100°	+150°,-60°
	Axis 3	+256°,-29°	±142.5°	+166°,-119°	+120°,-150°
	Axis 4	±190°	±270°	±190°	±360°
	Axis 5	±120°	+133.5°,-122.5°	±120°	±135°
	Axis 6	±360°	±270°	±360°	±360°
Controller		(called Smartcontroller) Ethernet TCP/IP interface	(CS8C) Ethernet, field bus interface	(RC7) Ethernet	(D42) Ethernet, Profibus-dp
real-time path modification (ALTER)		YES (requires V+ Extensions)	YES (VAL3 programme language)	No	No
Noise		NO figure for this	NO figure for this	NO figure for this	NO figure for this

APPENDIX B - Six-axis ATI GAMMA force/torque sensor

B-1 ATI Gamma F/T sensor with controller unit

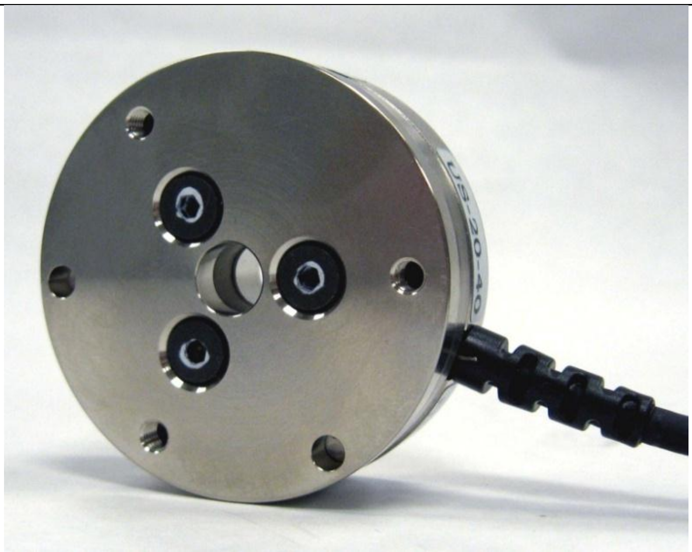
ATI INDUSTRIAL AUTOMATION
Force/Torque Sensor System Controller

B-2 ATI GAMMA F/T Sensor Specification

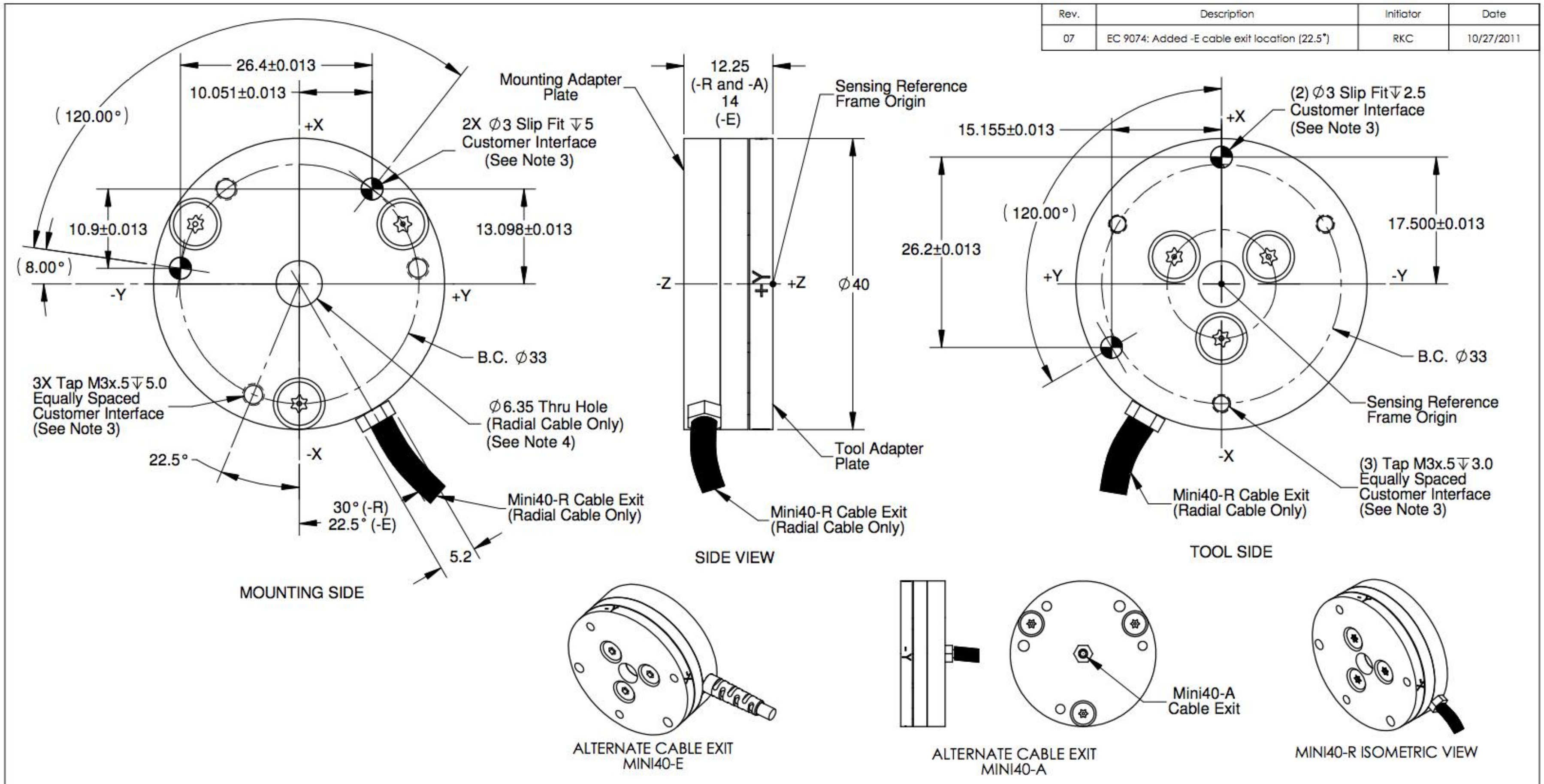
Single-Axis Sensing Ranges	
Force x and y	±130 N
Force z	±400 N
Torque x, y and z	±10 N-m
Resolution	
Fx & Fy	1/20 N
Fz	1/10 N
Tx,Ty & Tz	1/400 N-m
Stiffness	
Kx & Ky	9.1×10^6 N/m
Kz	1.8×10^7 N/m
Ktx & Kty	1.1×10^4 Nm/rad
Ktz	1.4×10^4 Nm/rad
Overload Protection	
Fx & Fy	±1200 N
Fz	±4100 N
Tx & Ty	±79 N-m
Tz	±82 N-m
Physical specification	
Weight	0.255 kg
Diameter	75 mm
Height	33 mm

APPENDIX C - ATI MINI40 Force/Torque DAQ System

C-1 Force/Torque transducer specification

<table>
<tr><td colspan="5">ATI MINI SI-40-2</td></tr>
<tr><td colspan="5">
</td></tr>
<tr><td colspan="5">Calibration Specifications</td></tr>
<tr><td>Axis</td><td>x,y</td><td>z</td><td>Rx, Ry</td><td>Rz</td></tr>
<tr><td>Sensing Ranges</td><td>40 N</td><td>120 N</td><td>2 Nm</td><td>2 Nm</td></tr>
<tr><td>Resolution</td><td>1/100 N</td><td>1/50 N</td><td>1/4000 Nm</td><td>1/4000 Nm</td></tr>
<tr><td>Single axis overload</td><td>±810 N</td><td>±2400 N</td><td>±19 Nm</td><td>±20 Nm</td></tr>
<tr><td>Stiffness</td><td>$1.1\text{x}10^7$ N/m</td><td>$2.0\text{x}10^7$ N/m</td><td>$2.8\text{x}10^3$Nm /rad</td><td>$4.0\text{x}10^2$ N/m</td></tr>
<tr><td>Resonant frequency</td><td>3200 Hz</td><td>4900 Hz</td><td>4900 Hz</td><td>3200 Hz</td></tr>
<tr><td colspan="5">Physical Specifications</td></tr>
<tr><td>Weight</td><td colspan="4">0.0499 kg</td></tr>
<tr><td>Diameter</td><td colspan="4">40 mm</td></tr>
<tr><td>Height</td><td colspan="4">12 mm</td></tr>
</table>

Rev.	Description	Initiator	Date
07	EC 9074: Added -E cable exit location (22.5°)	RKC	10/27/2011


26.4±0.013
10.051±0.013
(120.00°)
+X
Mounting Adapter Plate
2X ∅3 Slip Fit ↧5 Customer Interface (See Note 3)
10.9±0.013
13.098±0.013
(8.00°)
-Y
+Y
B.C. ∅33
3X Tap M3x.5↧5.0 Equally Spaced Customer Interface (See Note 3)
∅6.35 Thru Hole (Radial Cable Only) (See Note 4)
22.5°
-X
30° (-R)
22.5° (-E)
Mini40-R Cable Exit (Radial Cable Only)
5.2
MOUNTING SIDE
12.25 (-R and -A)
14 (-E)
Sensing Reference Frame Origin
-Z
+Z
+Y
∅40
Tool Adapter Plate
Mini40-R Cable Exit (Radial Cable Only)
SIDE VIEW
15.155±0.013
+X
(2) ∅3 Slip Fit↧2.5 Customer Interface (See Note 3)
(120.00°)
26.2±0.013
17.500±0.013
+Y
-Y
B.C. ∅33
Sensing Reference Frame Origin
-X
(3) Tap M3x.5↧3.0 Equally Spaced Customer Interface (See Note 3)
Mini40-R Cable Exit (Radial Cable Only)
TOOL SIDE
ALTERNATE CABLE EXIT MINI40-E
Mini40-A Cable Exit
ALTERNATE CABLE EXIT MINI40-A
MINI40-R ISOMETRIC VIEW

Notes:
1. Mounting and Tool Adapter made of either Aluminum or Stainless Steel (customer specified). Transducer made of hardened Stainless Steel.
2. **WARNING: DO NOT LOOSEN OR REMOVE INTERFACE PLATES OR CABLE FITTINGS DUE TO POTENTIAL DAMAGE.**
3. **DO NOT EXCEED INTERFACE DEPTH, MAY CAUSE DAMAGE.**
4. For Radial cable units: Do not allow items passing through center hole to contact transducer. Contact will affect reading accuracy.
5. Connector (not shown) has 17mm diameter and is 67.5mm long.

NOTES: UNLESS OTHERWISE SPECIFIED.

DO NOT SCALE DRAWING. ALL DIMENSIONS ARE IN MILLIMETERS.

3rd ANGLE PROJECTION

ATI INDUSTRIAL AUTOMATION

1031 Goodworth Drive, Apex, NC 27539, USA
Tel: +1.919.772.0115 Email: info@ati-ia.com
Fax: +1.919.772.8259 www.ati-ia.com
ISO 9001 Registered Company

PROPERTY OF ATI INDUSTRIAL AUTOMATION, INC. NOT TO BE REPRODUCED IN ANY MANNER EXCEPT ON ORDER OR WITH PRIOR WRITTEN AUTHORIZATION OF ATI.

DRAWN BY: D. Swanson 3/9/06

CHECKED BY: D. Perry 3/9/06

PROJECT # - SHEET 1 OF 1

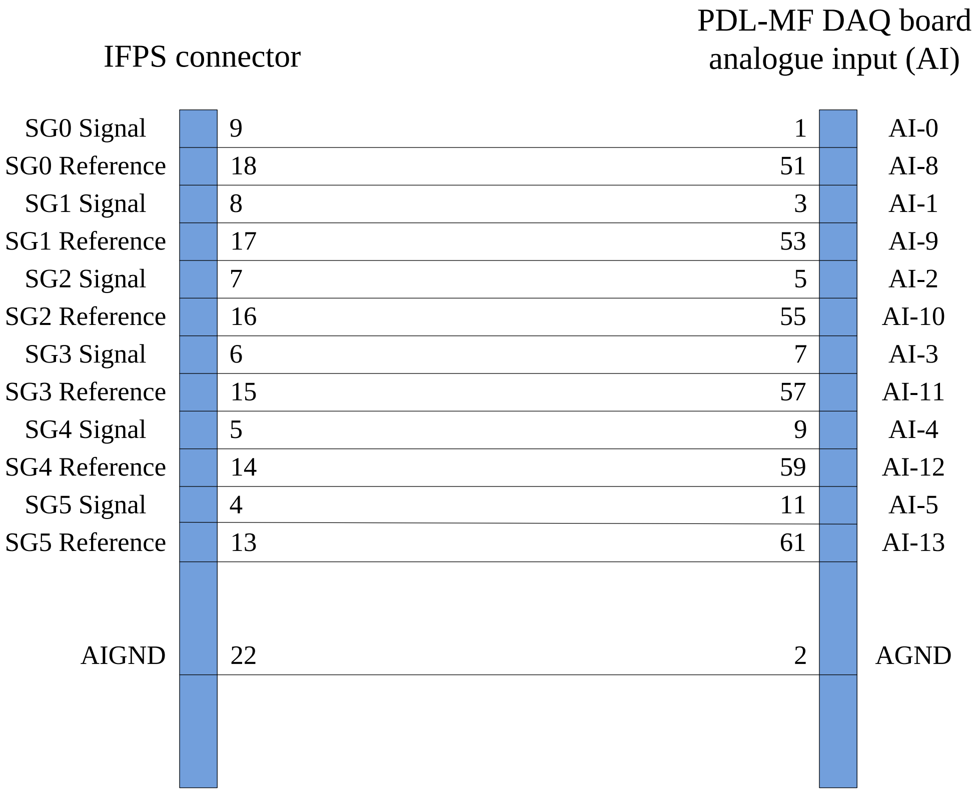
TITLE Mini40 Transducer Radial & Axial Cable Exit

SCALE	SIZE	DRAWING NUMBER	REVISION
2:1	B	9230-05-1278	07

C-2 Mini SI-40-2 Calibration matrix

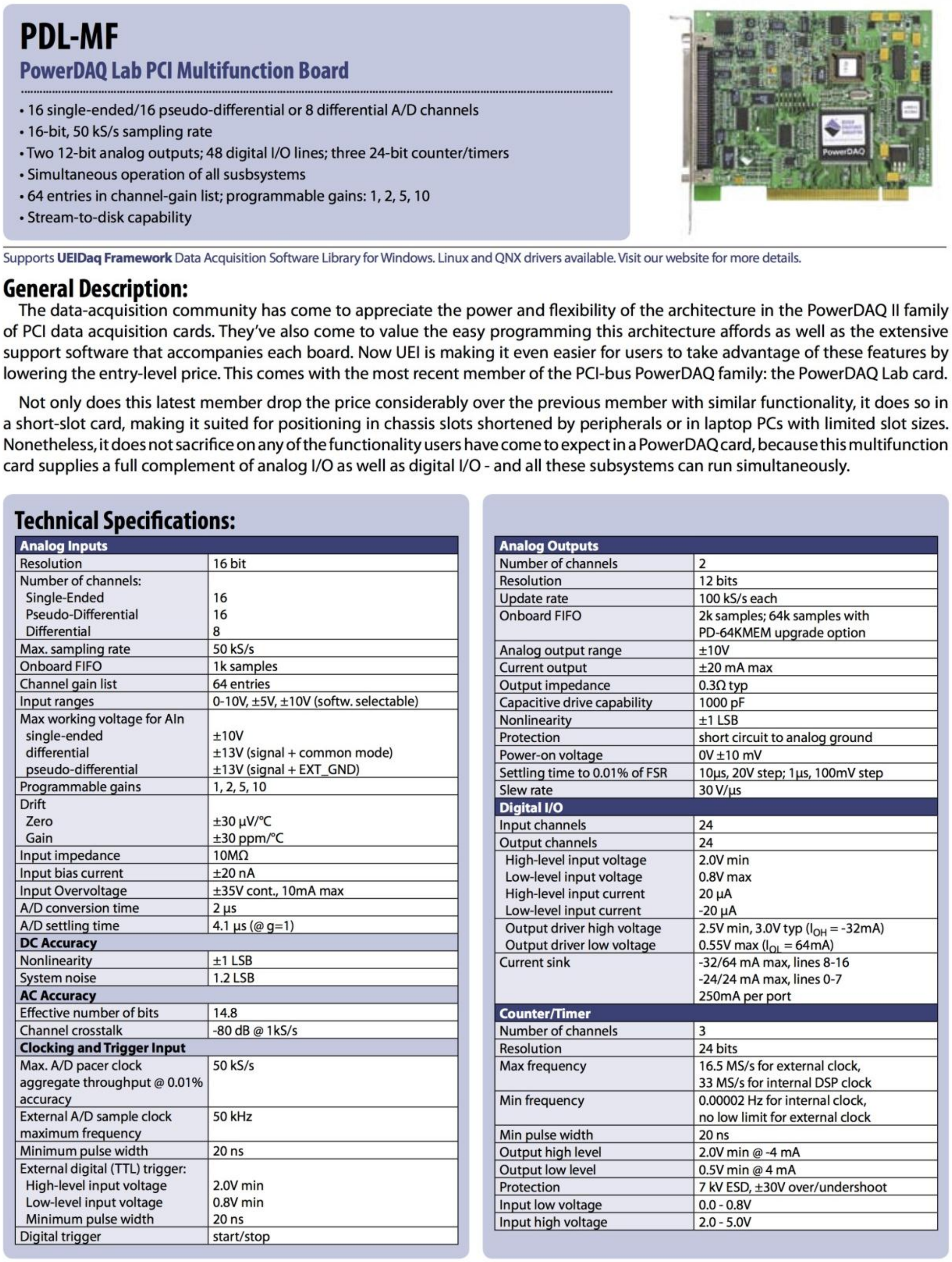
-0.20443	0.01529	3.70227	33.70273	-3.36040	-29.16939
-3.16047	-36.62555	1.56423	19.64245	2.36026	16.82625
18.26052	-1.21449	18.42587	-0.32543	18.71247	0.31949
-0.34942	-0.19484	31.57884	-0.67089	-32.71129	-0.33416
-37.36835	2.60569	17.56770	-0.54477	18.17552	0.49396
-1.77931	-18.41866	-1.58241	-19.12768	-1.95353	-17.13172

C-3 Transducer and PDL-MF differential input connection



IFPS connector		PDL-MF DAQ board analogue input (AI)	
SG0 Signal	9	1	AI-0
SG0 Reference	18	51	AI-8
SG1 Signal	8	3	AI-1
SG1 Reference	17	53	AI-9
SG2 Signal	7	5	AI-2
SG2 Reference	16	55	AI-10
SG3 Signal	6	7	AI-3
SG3 Reference	15	57	AI-11
SG4 Signal	5	9	AI-4
SG4 Reference	14	59	AI-12
SG5 Signal	4	11	AI-5
SG5 Reference	13	61	AI-13
AIGND	22	2	AGND

C-4 PDL-MF Board



PDL-MF

PowerDAQ Lab PCI Multifunction Board

- 16 single-ended/16 pseudo-differential or 8 differential A/D channels
- 16-bit, 50 kS/s sampling rate
- Two 12-bit analog outputs; 48 digital I/O lines; three 24-bit counter/timers
- Simultaneous operation of all susbsystems
- 64 entries in channel-gain list; programmable gains: 1, 2, 5, 10
- Stream-to-disk capability

Supports **UEIDaq Framework** Data Acquisition Software Library for Windows. Linux and QNX drivers available. Visit our website for more details.

General Description:

The data-acquisition community has come to appreciate the power and flexibility of the architecture in the PowerDAQ II family of PCI data acquisition cards. They've also come to value the easy programming this architecture affords as well as the extensive support software that accompanies each board. Now UEI is making it even easier for users to take advantage of these features by lowering the entry-level price. This comes with the most recent member of the PCI-bus PowerDAQ family: the PowerDAQ Lab card.

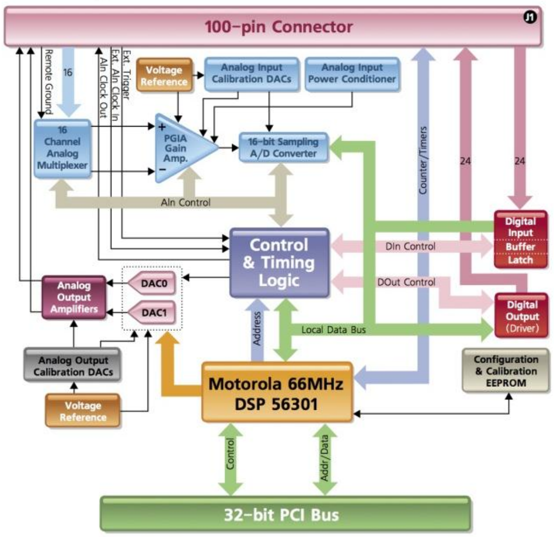
Not only does this latest member drop the price considerably over the previous member with similar functionality, it does so in a short-slot card, making it suited for positioning in chassis slots shortened by peripherals or in laptop PCs with limited slot sizes. Nonetheless, it does not sacrifice on any of the functionality users have come to expect in a PowerDAQ card, because this multifunction card supplies a full complement of analog I/O as well as digital I/O - and all these subsystems can run simultaneously.

Technical Specifications:

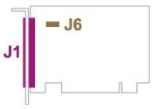
Analog Inputs	
Resolution	16 bit
Number of channels: Single-Ended Pseudo-Differential Differential	 16 16 8
Max. sampling rate	50 kS/s
Onboard FIFO	1k samples
Channel gain list	64 entries
Input ranges	0-10V, ±5V, ±10V (softw. selectable)
Max working voltage for AIn single-ended differential pseudo-differential	 ±10V ±13V (signal + common mode) ±13V (signal + EXT_GND)
Programmable gains	1, 2, 5, 10
Drift Zero Gain	 ±30 µV/°C ±30 ppm/°C
Input impedance	10MΩ
Input bias current	±20 nA
Input Overvoltage	±35V cont., 10mA max
A/D conversion time	2 µs
A/D settling time	4.1 µs (@ g=1)
DC Accuracy	
Nonlinearity	±1 LSB
System noise	1.2 LSB
AC Accuracy	
Effective number of bits	14.8
Channel crosstalk	-80 dB @ 1kS/s
Clocking and Trigger Input	
Max. A/D pacer clock aggregate throughput @ 0.01% accuracy	50 kS/s
External A/D sample clock maximum frequency	50 kHz
Minimum pulse width	20 ns
External digital (TTL) trigger: High-level input voltage Low-level input voltage Minimum pulse width	 2.0V min 0.8V min 20 ns
Digital trigger	start/stop

Analog Outputs	
Number of channels	2
Resolution	12 bits
Update rate	100 kS/s each
Onboard FIFO	2k samples; 64k samples with PD-64KMEM upgrade option
Analog output range	±10V
Current output	±20 mA max
Output impedance	0.3Ω typ
Capacitive drive capability	1000 pF
Nonlinearity	±1 LSB
Protection	short circuit to analog ground
Power-on voltage	0V ±10 mV
Settling time to 0.01% of FSR	10µs, 20V step; 1µs, 100mV step
Slew rate	30 V/µs
Digital I/O	
Input channels	24
Output channels	24
High-level input voltage Low-level input voltage High-level input current Low-level input current	2.0V min 0.8V max 20 µA -20 µA
Output driver high voltage Output driver low voltage	2.5V min, 3.0V typ (I_{OH} = -32mA) 0.55V max (I_{OL} = 64mA)
Current sink	-32/64 mA max, lines 8-16 -24/24 mA max, lines 0-7 250mA per port
Counter/Timer	
Number of channels	3
Resolution	24 bits
Max frequency	16.5 MS/s for external clock, 33 MS/s for internal DSP clock
Min frequency	0.00002 Hz for internal clock, no low limit for external clock
Min pulse width	20 ns
Output high level	2.0V min @ -4 mA
Output low level	0.5V min @ 4 mA
Protection	7 kV ESD, ±30V over/undershoot
Input low voltage	0.0 - 0.8V
Input high voltage	2.0 - 5.0V

Block Diagram:

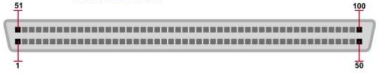


Pinout Diagrams:



J1 — 100-pin connector (female):

AIN8	1	51	AIN0
AGND	2	52	AGND
AIN9	3	53	AIN1
AGND	4	54	AGND
AIN10	5	55	AIN2
AGND	6	56	AGND
AIN11	7	57	AIN3
AGND	8	58	AGND
AIN12	9	59	AIN4
AGND	10	60	AGND
AIN13	11	61	AIN5
AGND	12	62	AGND
AIN14	13	63	AIN6
AGND	14	64	AGND
AIN15	15	65	AIN7
AGND	16	66	EXT_GND
AOUT0	17	67	AOUT1
AGND	18	68	AGND
DIN1	19	69	DIN0
DIN3	20	70	DIN2
DIN5	21	71	DIN4
DIN7	22	72	DIN6
DIN9	23	73	DIN8
DIN11	24	74	DIN10
DIN13	25	75	DIN12
DIN15	26	76	DIN14
DGND	27	77	DGND
DIN17	28	78	DIN16
DIN19	29	79	DIN18
DIN21	30	80	DIN20
DIN23	31	81	DIN22
DOUT1	32	82	DOUT0
DOUT3	33	83	DOUT2
DOUT5	34	84	DOUT4
DOUT7	35	85	DOUT6
DGND	36	86	+5VPJ2
DOUT9	37	87	DOUT8
DOUT11	38	88	DOUT10
DOUT13	39	89	DOUT12
DOUT15	40	90	DOUT14
DOUT17	41	91	DOUT16
DOUT19	42	92	DOUT18
DOUT21	43	93	DOUT20
DOUT23	44	94	DOUT22
DGND	45	95	DGND
EXT_TRIG_IN	46	96	TMR2
CV_OUT	47	97	DGND
EXT_TRIG_OUT	48	98	TMR1
CL_OUT	49	99	DGND
EXT_CLK	50	100	TMR0



J6 — 3-pin jumper header:

Connection Schemes:

Connector On The Board	Cable Required	Target Panel	Description
J1	PDL-CBL-100*	PDL-STP	Carries 16 analog input lines, 2 analog output lines, 24 digital input and 24 digital output lines, 3 counter/timer lines to terminal panel. PDL-STP allows further connection to 5B signal conditioning panel (for analog input signals).

* Pins 1 - 50 from PDL-MF's J1 connector are transfered to the first 50-pin IDC header of PDL-CBL-100 cable without remapping (pin-to-pin number match). Remaining 51 - 100 pins are transfered to the second 50-pin IDC header as follows: 51 to 1, 52 to 2, ... 99 to 49, 100 to 50.

Analog Input Configuration:

Configuration is performed using J6 jumpers.

J6 Jumper Position	Analog Input Configuration
3 2 1 ○ ○ ○	pseudo-differential
3 2 1 ○ ○ ○	single-ended
(default)	differential

C-5 Single scan mode data acquisition functions

pd_ain_set_config( ) – sets the analog input subsystem configuration: input mode, trigger and clock settings.

pd_ain_set_channel_list( ) – programs the ADC Channel List as required.

pd_ain_set_enable_conversion( ) – enables or disables AI conversions and permits complete the AI configuration before the subsystem response to the Start trigger set up.

pd_ain_sw_start_trigger( ) – triggers the AI Start event to start sample acquisition and should be on software mode.

pd_ain_sw_cl_start( ) – pulses the ADC Conversion Start signal.

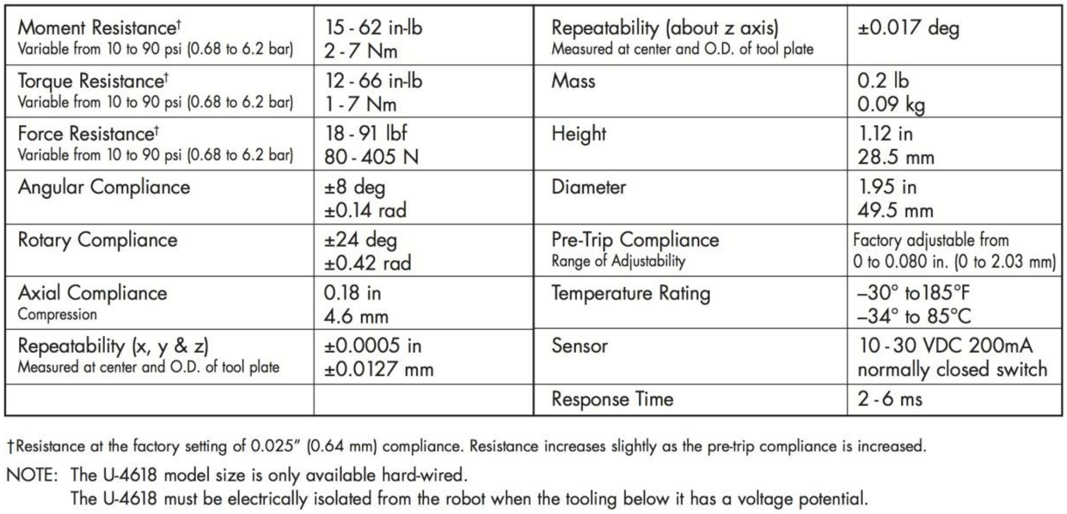
pd_ain_get_samples( ) – reads up to maximum buffer size samples from the ADC FIFO until it is empty. Each sample is stored in 16bits.

pd_ain_raw_to_volts( ) – converts data from raw values to volts.

pd_ain_clear_data( ) – clears the ADC FIFO and all AI data storage buffers.

APPENDIX D - Safety System Components

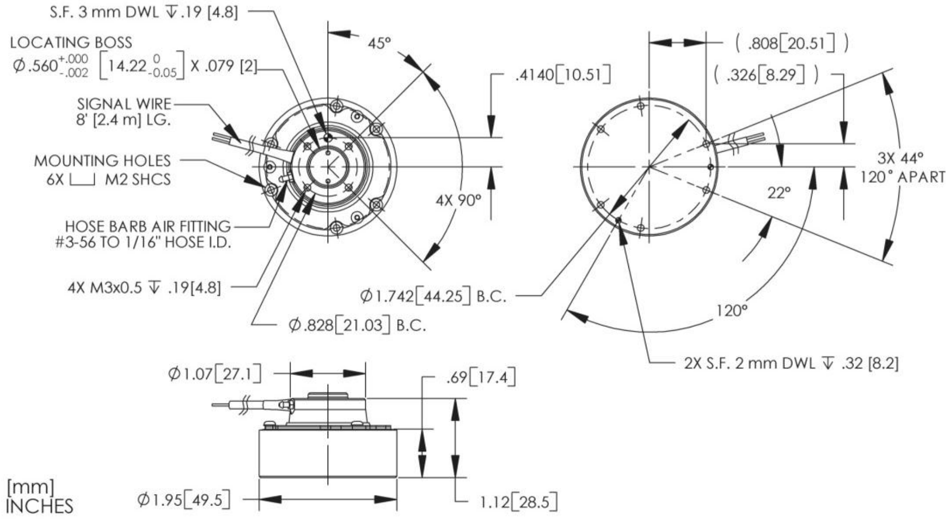
D-1 RAD Collision Sensor Model U-4618



Moment Resistance† Variable from 10 to 90 psi (0.68 to 6.2 bar)	15 - 62 in-lb 2 - 7 Nm	Repeatability (about z axis) Measured at center and O.D. of tool plate	±0.017 deg
Torque Resistance† Variable from 10 to 90 psi (0.68 to 6.2 bar)	12 - 66 in-lb 1 - 7 Nm	Mass	0.2 lb 0.09 kg
Force Resistance† Variable from 10 to 90 psi (0.68 to 6.2 bar)	18 - 91 lbf 80 - 405 N	Height	1.12 in 28.5 mm
Angular Compliance	±8 deg ±0.14 rad	Diameter	1.95 in 49.5 mm
Rotary Compliance	±24 deg ±0.42 rad	Pre-Trip Compliance Range of Adjustability	Factory adjustable from 0 to 0.080 in. (0 to 2.03 mm)
Axial Compliance Compression	0.18 in 4.6 mm	Temperature Rating	–30° to185°F –34° to 85°C
Repeatability (x, y & z) Measured at center and O.D. of tool plate	±0.0005 in ±0.0127 mm	Sensor	10 - 30 VDC 200mA normally closed switch
		Response Time	2 - 6 ms

†Resistance at the factory setting of 0.025" (0.64 mm) compliance. Resistance increases slightly as the pre-trip compliance is increased.

NOTE: The U-4618 model size is only available hard-wired.
The U-4618 must be electrically isolated from the robot when the tooling below it has a voltage potential.

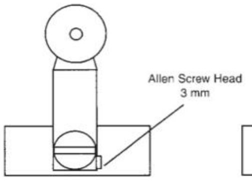


D-2 Honeywell Limit Switch

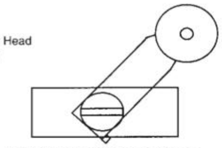
Side Rotaries

The side rotary assemblies incorporate a feature for adjusting the free position of the side rotary lever. The EN 50041 body style allows infinite adjustment and reclamp. The EN 50047 style allows clamping in 10° increments. See the following diagrams for details of the mechanism in each case.

Standard EN 50041 body style



Side Rotary Standard Free Position

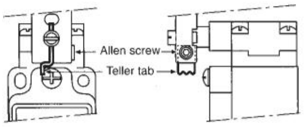


Side Rotary with Customised Free Position

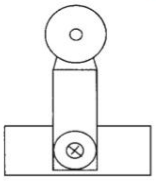
There are two lever mounting options: (1) By fully seating the lever in one of the four 90° detent positions on the shaft hub which provides positive lever retention; (2) By mounting the lever on the serrated portion of the shaft (which enables the lever to be mounted in any position).

To change the rotary lever's free position: (1) Use a 3 mm hex Allen wrench to loosen the Allen screw, as shown in the drawing above; (2) Back off the lever 2 mm and move it to the desired free position; (3) retighten the Allen screw; (4) Check to see if the free position is satisfactory for the application; (5) Repeat the adjustment procedure if necessary.

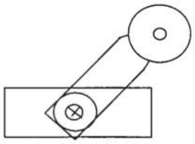
A teller tab located at the bottom of the lever (see diagram below) helps prevent lever slippage. It enables the installer to detect the correct tightening torque. When this tab cannot be moved, the Allen screw has been tightened properly.



Miniature EN 50047 body style



Side Rotary Standard Free Position



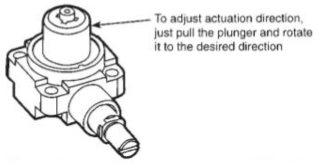
Side Rotary with Customised Free Position

A serrated coupling is used to set the lever free position in 10° increments. This adjustment is achieved by: (1) Unscrewing the combination head screw which holds the lever in place, taking care not to lose any parts; (2) Readjust the assembly and rotate to the desired free position; (3) Re-assemble and tighten the combination screw. (4) Check that the free position is correct for the application and repeat the adjustment procedure if necessary.

NOTE: The lever can be set in 90° increments by removing the lever and rotating it to the desired 90° position.

EN 50041 Side Rotary Actuator Direction Adjustment

As furnished, GLS rotary switches will operate when the lever is rotated from either the left or right. They can be field modified to operate in one direction only (Clockwise CW; Counter clockwise CCW) by following these steps: (1) Carefully remove the complete head assembly; (2) Turn the head assembly upside down as shown in the drawing below.


To adjust actuation direction, just pull the plunger and rotate it to the desired direction

(3) Pull the plunger mechanism out and rotate it through 90° degree increments until the alignment tab points to the desired function (CW, CCW, or CW and CCW). (4) Push plunger mechanism in. (5) Reassemble the head assembly and re-test the switch in your application.

Replacement Instructions

All levers for side rotaries are available as replacment parts. All basics, except the plug-in, can be replaced. All EN 50041 heads can be replaced. The replacement procedures for these components are straightforward in nature.

Side Rotary Levers

Remove the old lever from the product being replaced. On EN 50041 product this is achieved by loosening the Allen screw holding the lever on the shaft. On EN 50047 product this is achieved by unscrewing the combination screw holding the lever on the shaft.

Replace the lever and tighten the Allen screw or combination screw. Retest the switch in its application.

Heads

All EN 50041 style switch heads can be removed and replaced.

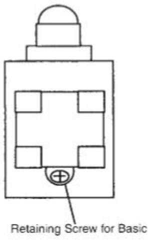
Remove the old head by unscrewing the four retaining screws on the head assembly.

Ensure replacement part is identical to one being removed.

Re-test the assembly and ensure correct operation.

Basics

Non plug-in EN 50041 and three conduit EN 50047 body styles. Basic switches can be removed and replaced by following this procedure: (1) Remove the cover from the body; (2) Before disconnecting the switch wiring, carefully note the wiring arrangement for your application, particularly the safety ground connection; (3) Remove the basic switch retaining screw; (4) Remove the old basic and replace it with the same thing; (5) Use the retaining screw to install the new basic – ensure that it is correctly seated in the switch body; (6) Wire the switch terminals as before; (7) Before replacing the cover – ensure that the switch wires are not twisted or otherwise lifted from the basic (to prevent them from becoming trapped when the cover is replaced); then (8) Test the switch in the application.



APPENDIX E - Test Programme

E-1 Evaluation of F/T controller and Po-ngaen's DAQ system

A set of experiments was designed to evaluate the performance of F/T DAQ system and quality of the acquired F/T signals in three difference conditions namely; 1) robot arm and robot controller power off, 2) robot arm power off and robot arm power on, 3) robot arm and robot controller power on. Data were collected using two F/T system interfaces without any applied force on the transducer as the true value is assumed to be zero; the ATI F/T controller and 12-bit F/T DAQ system developed by Po-ngaen [2006]. During the experiment the sensor is mounted on the end-effector and the robot arm is in a position where the joint 3 is parallel to the z-axis (zero position).

Table E-1 and Table E-2 compare the results obtained from signal's mean and standard deviation of the experimental data for both of the system interfaces. The results represent two general type of error that the system experience; accuracy and precision. First, the accuracy of measurement is expressed by the mean value that represents the amount of shift value from the true value. Similarly, the standard deviation is related to the precision of measurement as the value describing how much variation occurs between successive readings.

It can be seen from the data in Table E-2 that the DAQ system is tend to have poor repeatability particularly at *z* axis and significant random variation of the signals is clearly occurred when the robot controller and arm is powered on. The results of the experiment indicate that the DAQ system delicately affected by the environment. This discrepancy could be attributed to the interference robot arm mechanical structure as well as gravity effects that act parallel to *z* axis that cause dynamic effect to the robot arm when carrying the sensor. However, there were no significant differences were found for F/T controller output readings. Following this, the fluctuation/variation can be reduced by introducing a suitable filter, for example a moving average filter can be useful to reduce the effects of mechanical vibrations and inertia as well as improve the measurement precision.

Table E-1 F/T controller signal’s error mean and standard error deviation calculation

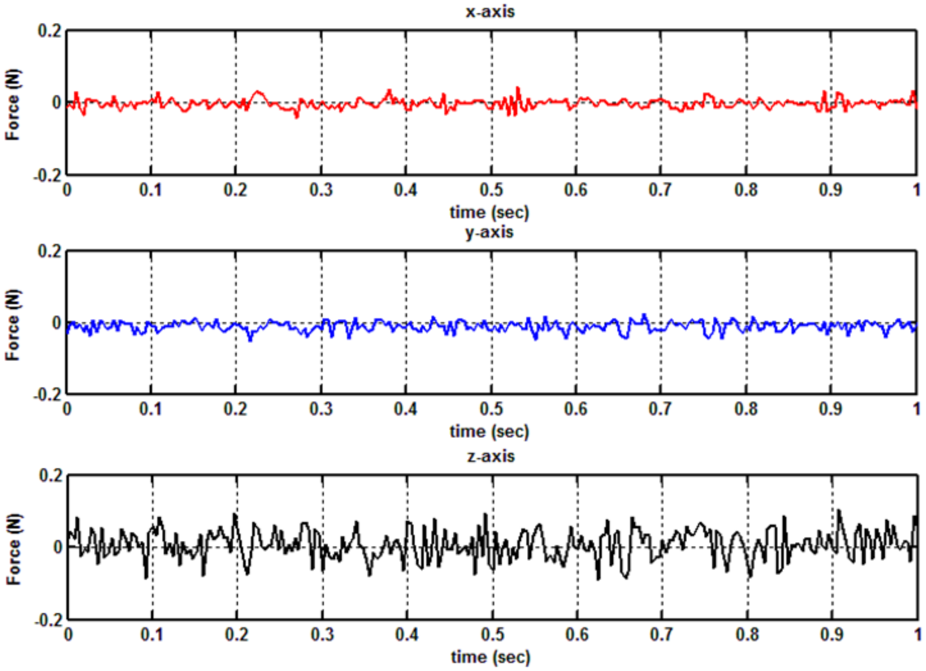
Axis	Robot controller & arm power off		Robot controller power on		Robot controller & arm power on	
	μ	σ	μ	σ	μ	σ
x	-1.856	0.3571	-0.5000	0.5689	-0.8654	0.3416
y	0.534	0.6883	0.7885	0.5495	0.8558	0.3781
z	-0.082	0.2727	-0.9885	0.1516	-0.1673	0.3736
rx	-0.428	0.4987	-0.9769	0.1503	-0.8692	0.3431
ry	-0.254	0.2757	-0.72301	0.4479	-0.5039	0
rz	0.274	0.5150	-0.0827	0.2757	-0.0077	0.1072

Table E-2 DAQ signal’s error mean and standard error deviation calculation

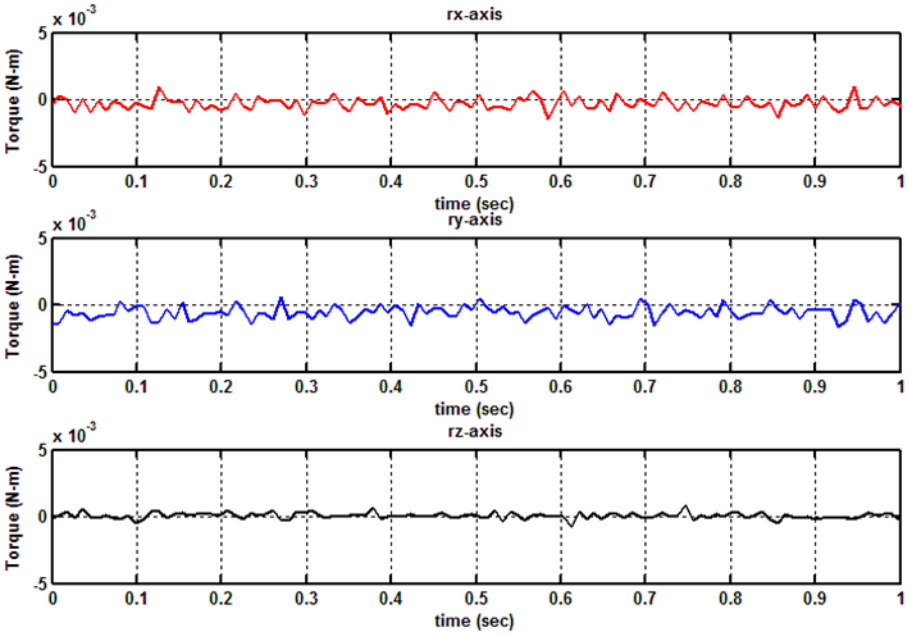
Axis	Robot controller & arm power off		Robot controller power on		Robot controller & arm power on	
	μ	σ	μ	σ	μ	σ
x	-0.0019	0.0492	0.0001	0.0267	0.1368	1.0100
y	0.0054	0.0735	0.0057	0.0782	0.2059	0.9992
z	-0.2578	0.5079	0.0257	0.4777	1.7324	2.2001
rx	0.0001	0.0207	0	0.0239	0.0022	0.8740
ry	0.0001	0.0615	0.0004	0.0609	0.4421	1.0533
rz	0.7615	0.4267	0.6707	0.4706	0.6022	0.8924

E-2 Mini40 DAQ system evaluation

Mode (1) Robot arm and controller power off

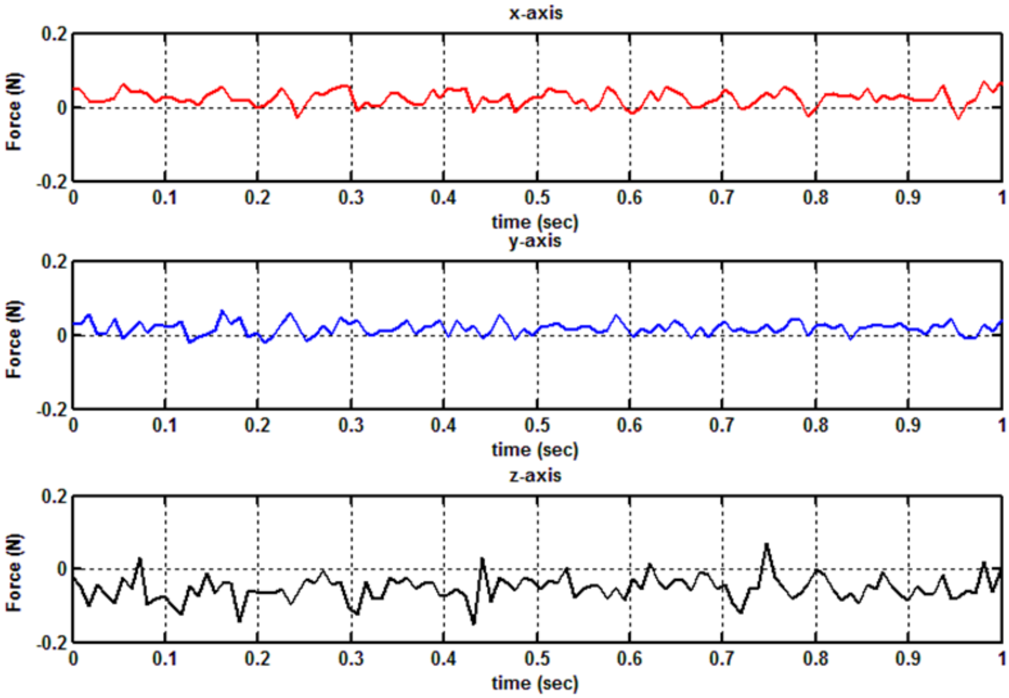


Force data

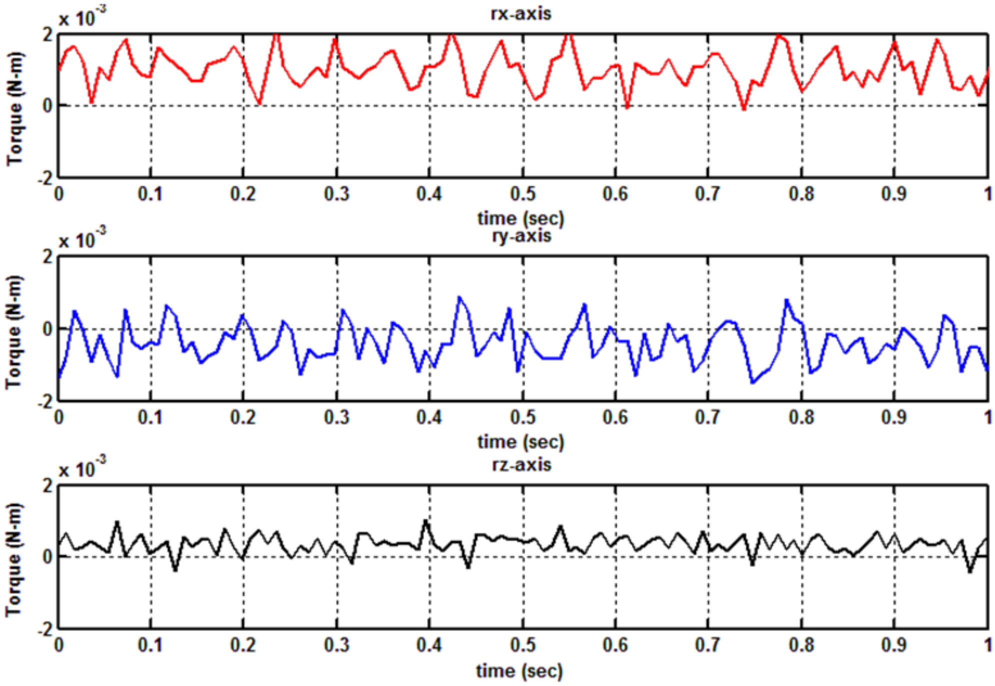


Torque data

Mode (4) ALTER mode



Force data



Torque data

E-3 Ethernet socket communication evaluation

An experiment was carried in order to guarantee the reliability of the socket communication between host PC and robot controller via socket TCP protocol at 100Mbit/s as shown in Figure E-3-1.

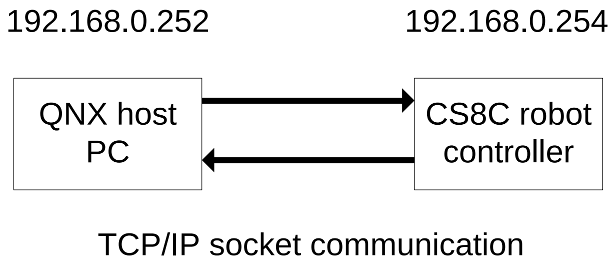


Figure E-3-1 TCP/IP socket communication

A set of single axis (x-axis) sample trajectory is generated by the host PC and transmitted to the robot controller as schematically shown in Figure E-3-2. The incremental position is sent every 4ms and repeated until the final incremental value.

First the robot moves to the start position (point A) and then to point B by using '*alter move'* command. The alter mode is activated when the robot arrives at point B by setting the activation flag. The trajectory between point B and point C is based on position data received from host PC in real time. The movement strategy is shown in Figure E-3-3.

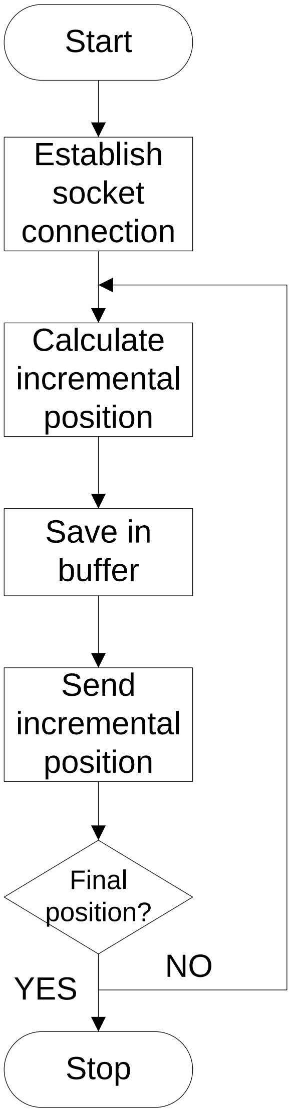

Start
Establish socket connection
Calculate incremental position
Save in buffer
Send incremental position
Final position?
NO
YES
Stop

Figure E-3-2 The experiment flowchart to evaluate communication time and dynamic response of manipulator arm

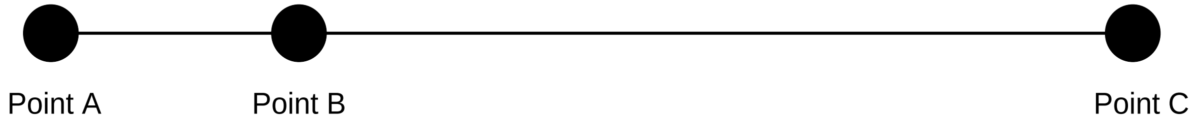


Figure E-3-3 The Cartesian alteration strategy

The example of VAL3 code program below shows the ALTER feature modifies the robot's position with respect to its start position.

```
begin
//Moves to start position
movej(ptA,flange,mMotionMDesc)
//MoveAlter
alterMovel(ptB,flange,mMotionMDesc)
//Waits until robot arrive at point B
//to start alter. The trajectory in
//between point A are B cannot be
//altered.
waitEndMove()
bStartAlter=true
//When robot is at point B, alter begins
//The trajectory from point B to point C
//is done 100% by alter.
do
ptTemp=here(flange,world)
until ((ptTemp.trsf.x-ptC.trsf.x)&lt;0.5)
bFinishAlter=true
movel(ptC,flange,mMotionMDesc)
waitEndMove()
bDone=true
end
```

Test results for task varying robot maximum velocity of 100mm/s and 20mm/s is shown in following figures.

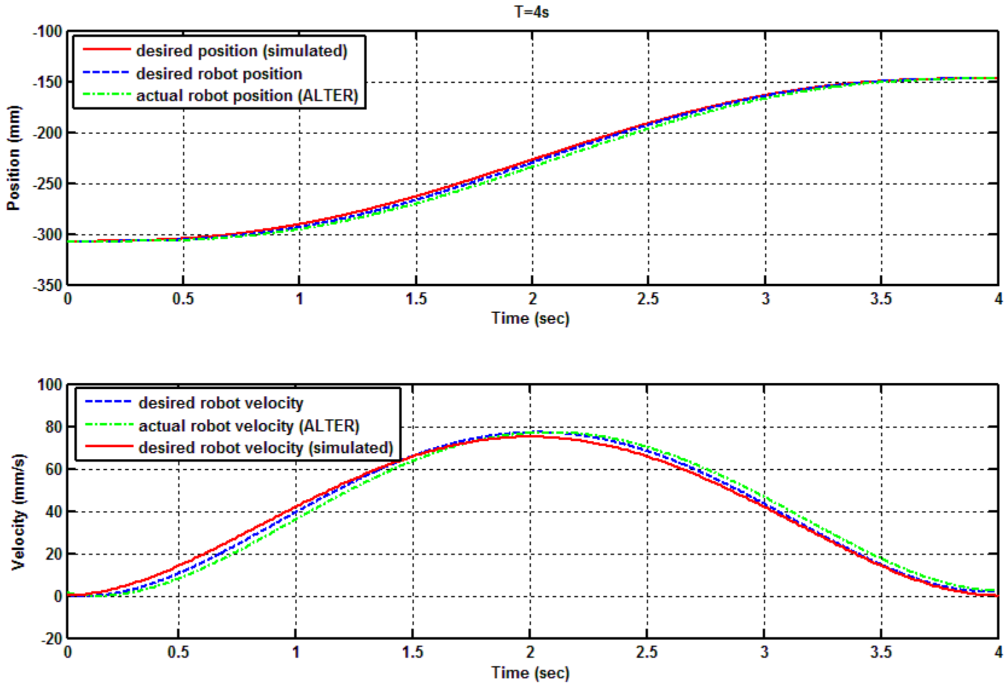


Figure E-3-4 Maximum velocity = 100mm/s

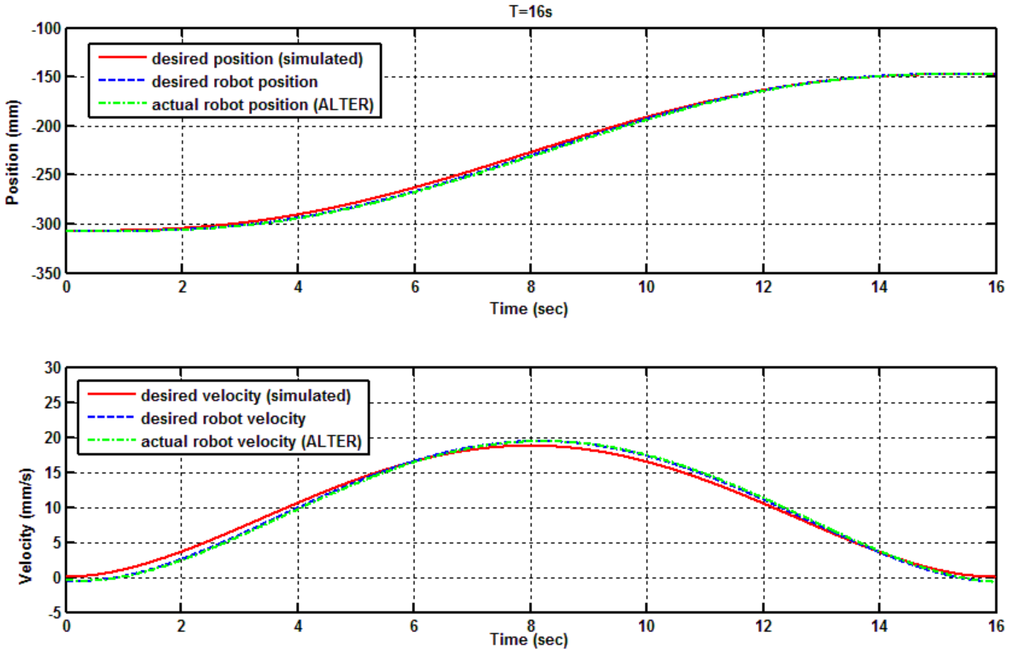


Figure E-3-5 Maximum velocity = 20mm/s

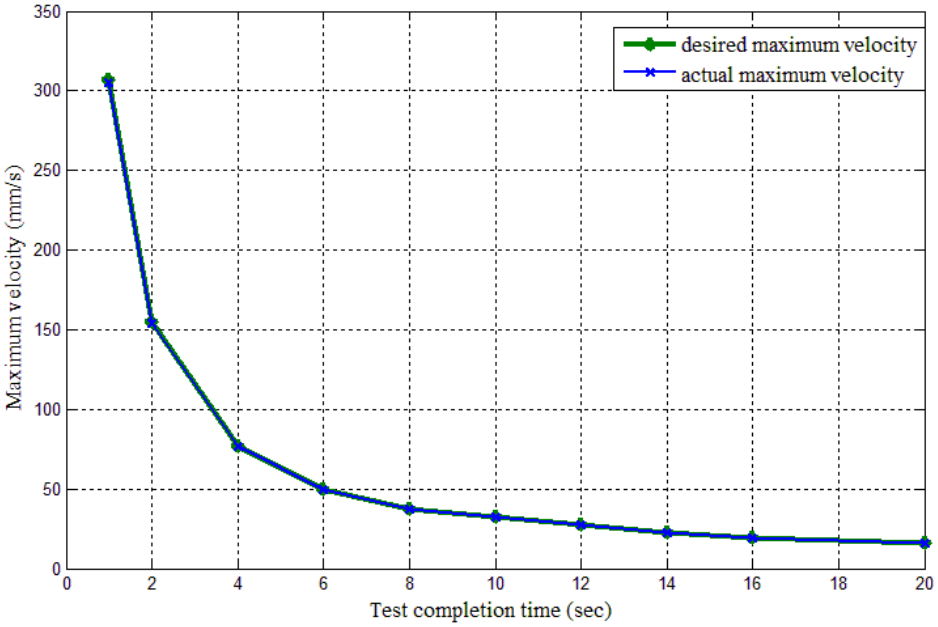


Figure E-3-6 ALTER maximum velocity vs. test completion time

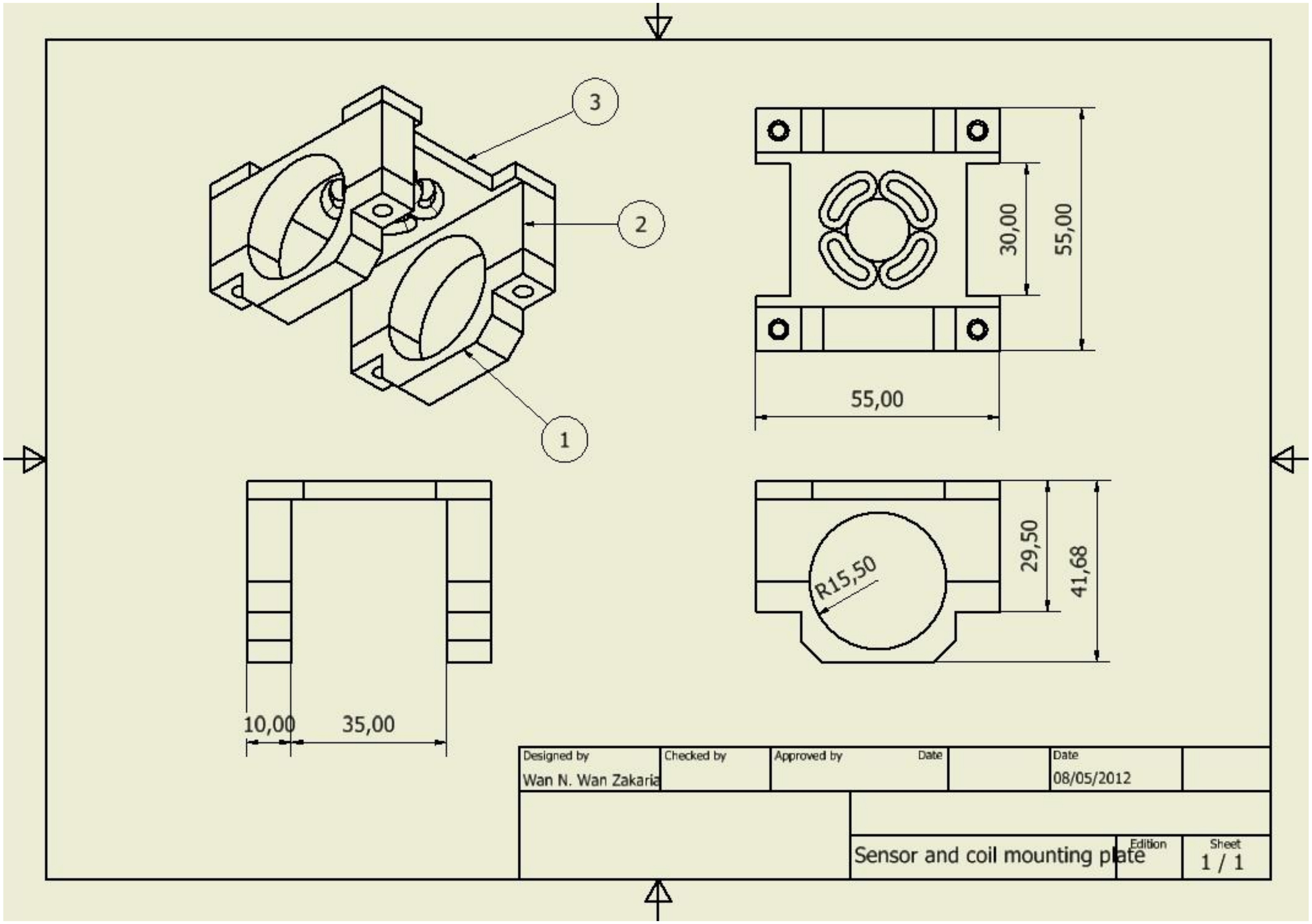
APPENDIX F – LAYOUT DRAWINGS

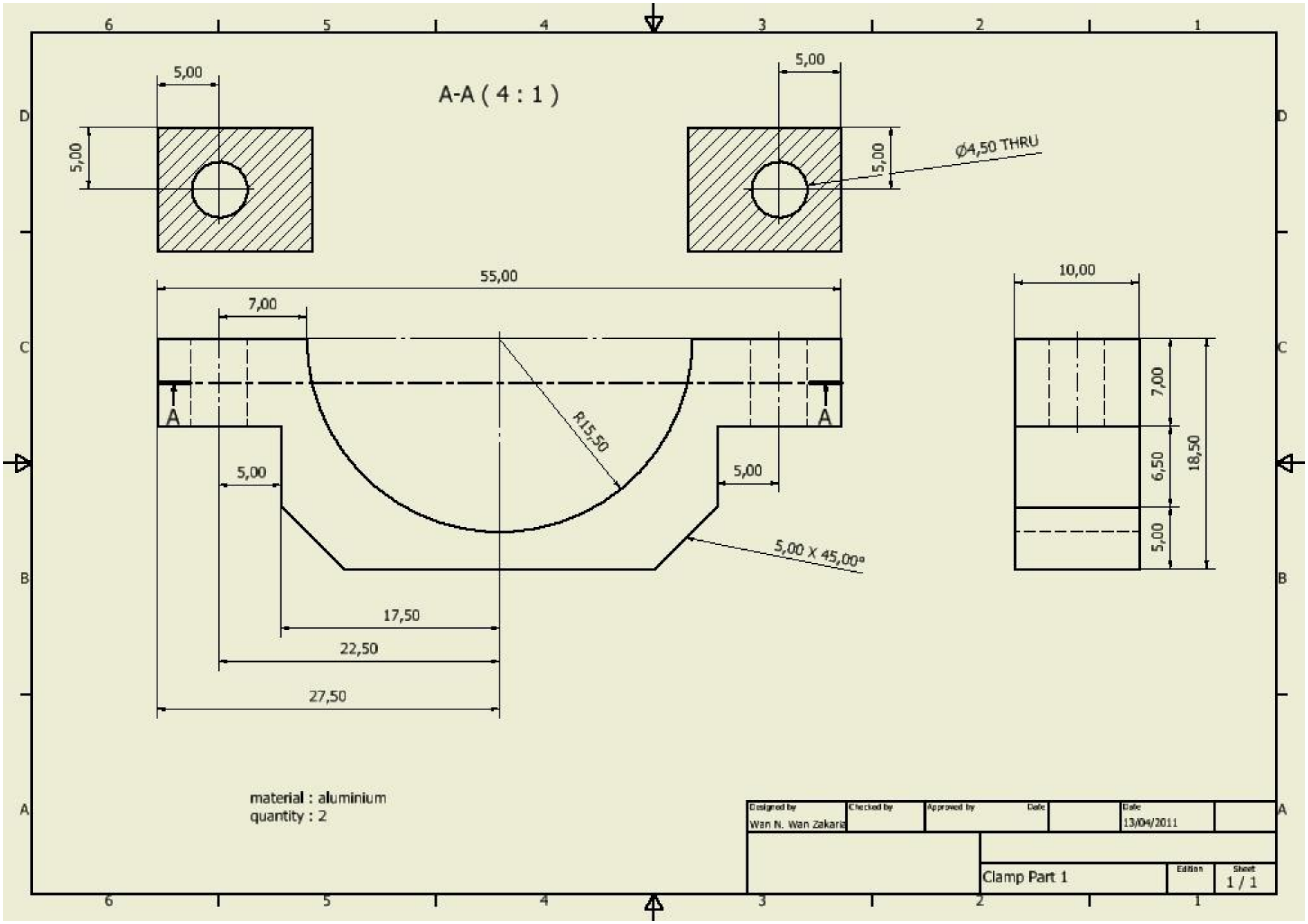
F-1 Sensor and Coil Assembly Drawing

F-2 Clamp Part 1

F-3 Clamp Part 2

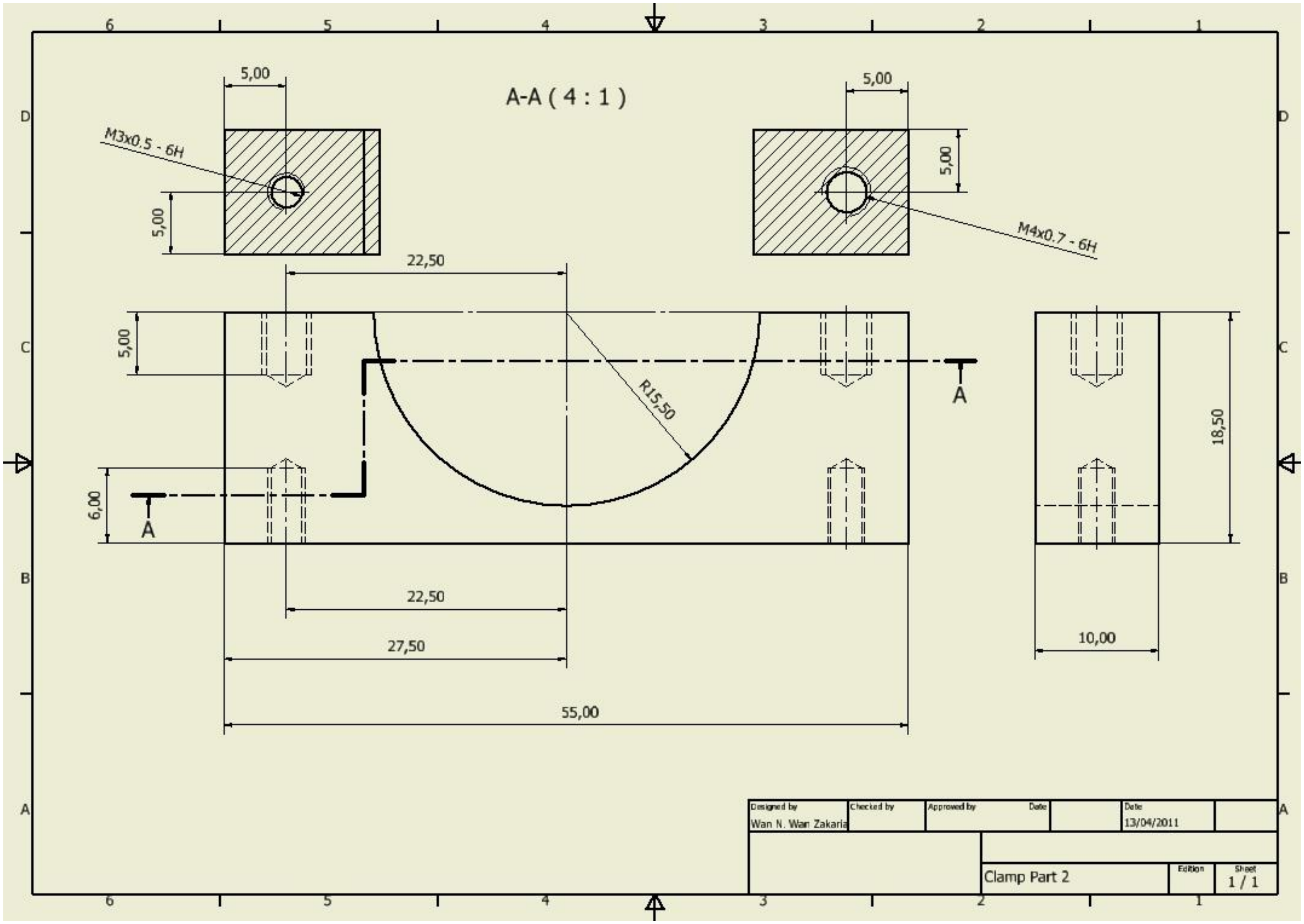
F-4 Sensor Adapter Plate


3
2
1
30,00
55,00
55,00
29,50
41,68
R15,50
10,00
35,00
Designed by
Wan N. Wan Zakaria
Checked by
Approved by
Date
Date
08/05/2012
Sensor and coil mounting plate
Edition
Sheet
1 / 1



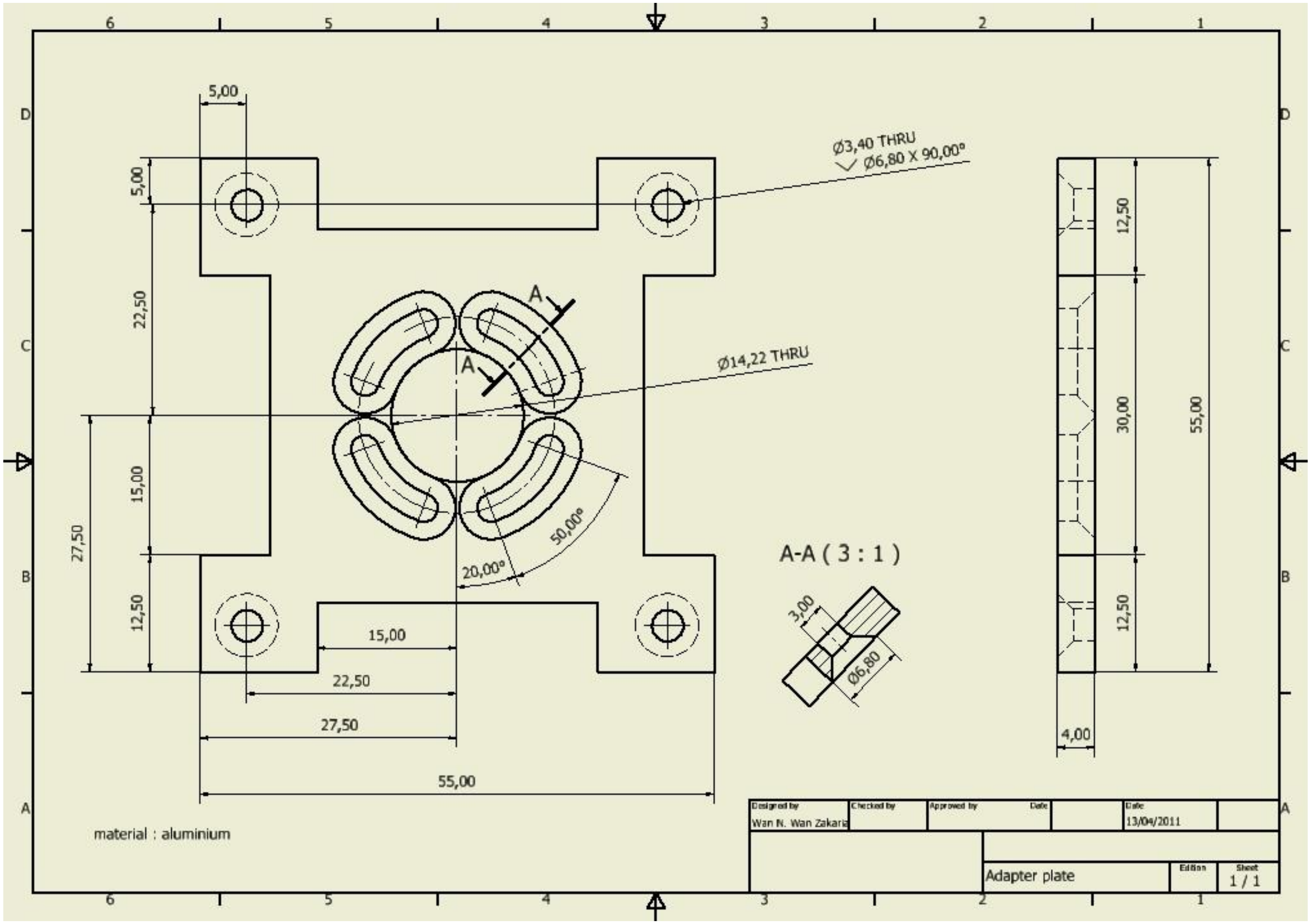
material : aluminium
quantity : 2

Designed by Wan N. Wan Zakaria	Checked by	Approved by Date		Date 13/04/2011	
		Clamp Part 1		Edition	Sheet 1 / 1



Designed by	Checked by	Approved by	Date	Date
Wan N. Wan Zakaria				13/04/2011

	Edition	Sheet
Clamp Part 2		1 / 1


5,00
5,00
22,50
15,00
27,50
12,50
Ø3,40 THRU
⌵ Ø6,80 X 90,00°
Ø14,22 THRU
A
A
50,00°
20,00°
15,00
22,50
27,50
55,00
A-A (3 : 1)
3,00
Ø6,80
12,50
30,00
55,00
12,50
4,00
material : aluminium
Designed by
Wan N. Wan Zakaria
Checked by
Approved by
Date
Date
13/04/2011
Adapter plate
Edition
Sheet
1 / 1
6
5
4
3
2
1
D
C
B
A