AUGUST 2016 Ph.D. in MECHANICAL ENGINEERING AHMED RAHMAN JASIM AL MUSAWI

UNIVERSITY OF GAZIANTEP
GRADUATE SCHOOL OF
NATURAL & APPLIED SCIENCES

IMPLEMENTATION OF LEARNING MOTION TO CONTROL A ROBOTIC ARM USING HAPTIC TECHNOLOGY

Ph.D. THESIS
IN
MECHANICAL ENGINEERING

BY
AHMED RAHMAN JASIM AL MUSAWI

AUGUST 2016

Implementation of Learning Motion to Control a Robotic Arm Using Haptic Technology

Ph.D. Thesis

in

Mechanical Engineering

University of Gaziantep

Supervisors

Prof. Dr. L. Canan DÜLGER

Prof. Dr. Sadettin KAPUCU

By

Ahmed Rahman Jasim AL MUSAWI

August 2016

© 2016 [Ahmed Rahman Jasim AL MUSAWI]

REPUBLIC OF TURKEY
UNIVERSITY OF GAZIANTEP
GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES
MECHANICAL ENGINEERING

Name of the thesis: Implementation of Learning Motion to Control a Robotic Arm Using Haptic Technology
Name of the student: Ahmed Rahman Jasim AL MUSAWI
Exam date: 26.08.2016

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Necmeddin YAZICI
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Mehmet Sait SÖYLEMEZ
Head of Department

This is to certify that we have read this thesis and that in our consensus/majority opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Sadettin KAPUCU
Co-Supervisor

Prof. Dr. L. Canan DÜLGER
Supervisor

Examining Committee Members:	Signature
Prof.Dr. Hakan YAVUZ	....
Prof.Dr.Sedat BAYSEÇ	....
Prof.Dr.L.Canan DÜLGER	....
Assist Prof.Dr. Tolgay KARA	....
Assist.Prof.Dr.M.Taylan DAŞ	....

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Ahmed Rahman Jasim AL MUSAWI

ABSTRACT

IMPLEMENTATION OF LEARNING MOTION TO CONTROL A ROBOTIC ARM USING HAPTIC TECHNOLOGY

AL MUSAWI, Ahmed Rahman Jasim
Ph.D. in Mechanical Engineering
Supervisors: Prof. Dr. L. Canan DÜLGER
Prof. Dr. Sadettin KAPUCU

August 2016, 131 pages

Robotic arm motion learning is one of the most important recent robotics research area which brings robots in human life. This study is divided into three parts; a new approach for Human Robot Interaction (HRI) is proposed by assisting haptic technology hypothesis, a novel inverse kinematics solution for robotic arm is designed based on Artificial Neural Network (ANN), and motion control is implemented for Robotic Assisted Thoracoscopic Surgery (RATS) based on the kinematics of Remote Center of Motion (RCM). A new controller structure has been introduced by using virtual spring control method by compensating tool inertia effect. The applied force and torque are transformed to the desired position/orientation through the simultaneous matching between the human direct guidance and robot response. The novelty of the proposed ANN is that of including the current joint angles configuration as well as the desired position and orientation in the input pattern of ANN. The traditional ANN has got only the desired position and orientation of the end effector in the input pattern of ANN. The motion of surgical robot is constrained by the kinematics of RCM, so a new control design for RCM is introduced. The controller is implemented to satisfy the requirement of RATS. The control method is then verified. The comprehensive experimental results have shown significant improvement in learning performance and reducing motion errors. The inclusion of current joint angles configuration in ANN significantly increased the accuracy of estimation of the joint angles output. The results have proved the applicability and the efficiency of the proposed design in robotic surgery, especially in RATS.

Keywords: Robotic arm, motion control, haptic technology, artificial neural network (ANN), robotic surgery, robotic assisted Thoracoscopic surgery (RATS)

ÖZET

HAPTİK TEKNOLOJİ KULLANARAK ROBOT KOLUNUN DENETİMİNDE ÖĞRENMENİN UYGULANMASI

AL MUSAWI, Ahmed Rahman Jasim
Doktora Tezi, Makine Mühendisliği
Danışmanlar: Prof. Dr. L. Canan DÜLGER
Prof. Dr. Sadettin KAPUCU

Ağustos 2016, 131 sayfa

Son yıllarda robot kolu ile hareket öğrenimi robotları insan yaşamında kaçınılmaz kılan en önemli araştırma alanlarından birisi olmuştur. Sunulan çalışma üç kısımda incelenmiştir; Haptik teknoloji destekli insan robot etkileşiminde yeni bir yaklaşımdır, Yapay Sinir Ağları (YSA) tabanlı tasarlanmış yeni(modern) ters kinematik çözümü ve RCM kinematiği tabanlı robot destekli toraskopik cerrahi için hareket denetiminin yapılmasıdır. Çalışma içeriğinde yer alan denemeler eğitim amaçlı kullanılmak üzere yapılmıştır. Uç işlemci atalet etkisini telafi eden sanal yay denetim metodu kullanılarak yeni bir denetim yapısı sunulmaktadır. Uygulanan kuvvet ve tork doğrudan insan-robot arasındaki anlık eşleştirmelerle istenilen konum ve yönelime dönüştürülmektedir. Geleneksel YSA'da sadece uç işlemcinin konum ve yönelimi YSA girdi örüntüsü olarak verilirken, uç işlemcinin konum ve yöneliminin yanı sıra anlık mafsal açılarının kullanılması önerilen YSA'nın yenilikçi kısmıdır. Cerrahi robot hareketi RCM kinematiği ile kısıtlanmış ve RCM için yeni bir denetim tasarımı sunulmuştur. RATS gerekliliğinin sağlanması adına denetleyici uygulanarak denetim yöntemi doğrulanmıştır. Kapsamlı deney sonuçları öğrenme performansında ve hareketin hatasının azaltılmasında gelişmeler olduğunu göstermektedir. YSA içerisine anlık mafsal açılarının eklenmesi çıktı mafsal açılarının tahminindeki doğruluğu ciddi anlamda arttırmıştır. Sonuçlar özellikle hedef alınan robotik cerrahi için sunulan tasarımın verimliliğini ve uygulanabilirliğini kanıtlar niteliktedir.

Anahtar Sözcükler: robot kolu, hareket denetimi, haptik teknoloji, yapay sinir ağları (YSA), robotik ameliyat, robot destekli toraskopik ameliyat (RDTA)

To My Parents, and My Family

ACKNOWLEDGMENTS

The author wishes to express his deepest gratitude to his supervisors Prof. Dr. L. Canan DÜLGER and Prof. Dr. Sadettin KAPUCU for their guidance, advice, criticism, encouragements and insight throughout the research.

The author would also like to thank Prof. Dr. Maruf Şanlı for his valuable assistance, suggestions and support during this study. (Department of Chest surgery, Faculty of Medicine, Gaziantep University).

This study is supported and granted by Gaziantep University/Scientific Research Projects Governing Unit (BAPYB). The project's code number is MF.14.02. The website 'http: //bap.gantep.edu.t' includes all transactions purchasing, reports, final reports and publications for this study. The author would like to thank them for their help and support.

I would like to acknowledge my sponsor, Turkish government scholarships / ministry of higher education, who gave me the opportunity to study a Ph.D. by granting me this scholarship. I would also like to thank the Ministry of Higher Education/ Iraq for the complementary scholarship I received during my studies.

Finally, with great love and respect, the author would deeply appreciate the continuous and strong support of his parents, wife, and brothers.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Page

LIST OF SYMBOLS/ABBREVIATIONS

$A_1\ A_2\ A_3\ A_4\ A_5\ A_6$	Homogeneous transformation matrices
C_{vs}	Compliance of virtual spring
F_{app}, T_{app}	Vectors of the applied force/torque acting
$F^s\ , T^s$	Vectors of the resultant force on F/T sensor
K_{vs}	Stiffness of the virtual spring
P_6^0	Position vector
P_{fg}	Position vector of trocar point from the wrist of the robot in wrist coordinate.
P_{fg_x}	Thoracoscope length in x-axis
P_{fg_y}	Thoracoscope length in the y-axis
P_{fg_z}	Thoracoscope length in the x-axis
$P_{insertion}$	Inserted part
R_6^0	Rotation matrix
a_j	Distance along x axis from origin to the intersection of x and z axis
ANN	Artificial Neural Network
DC	Direct current
DH	Denavit-Hartenberg
d_j	Offset along robot link
DOF	Degree of Freedom
E	Elasticity modulus
e	Strain
Enc	Joint encoder angles.
F/T	Force/Torque
F_g, T_g	Force and Torque caused by the inertia

FQA	Factored Quaternion Algorithm
GA	Genetic Algorithm
GUIDE	Graphical user interface development environment
HOL	High-order-logic
HRI	Human-Robot Interaction
I/O	Input / Output
ICSI	Intra-Cytoplasmic Sperm Injection
KF	Kalman Filter
MIS	Minimally Invasive Surgery
MLP	Multi-layered perceptron
MSE	Mean square error
NNs	Neural Networks
PID	Proportional-Integral-Derivative
$q, \dot{q}, \ddot{q}$	Joint angle, velocity, and acceleration
Q, QD, QDD	Trajectory joint angles, velocities and accelerations
$q_{Feedback}$	Current joint angles
RAMIS	Robotics assisted minimally invasive surgery
RCM	Remote center of motion
s	Stress
T_m	Joint motor torques
U	Joint vector of torque
Ufx, Ufy, Ufz	Raw signals of F/T sensor
VATS	Video-assisted Thoracoscopic surgery
VRM	Virtual Reality Model
VRML	Virtual Reality Modeling Language
VSD	Virtual spring damper
α_j	Angle about x axis
θ_j	Angle about z axis
$C(q, \dot{q})$	Coriolis and centripetal coupling matrix
$F(\dot{q})$	Joint friction force
$G(q)$	Gravity loading
$J(q)$	Jacobian matrix
$M(q)$	Robot inertia matrix

g	Joint forces due to a wrench
δP	Displacement of the virtual spring

CHAPTER 1

INTRODUCTION

1.1 Background of Robotic Arm

Robot is a Mechatronic intelligent agent that can carry out tasks with minimal supervision. Usually an electronic and mechanical devices are controlled by computer software or electronic embedded circuits [1]. Robots are used in a wide range of applications, industrial, medical, dangerous tasks and exploring outer space or under the sea [2]. Figure 1.1 shows medical application of robotic arm. Robotic types are manipulator robotic arms and mobile robots, wheeled robots, swimming robots, and flying robots.

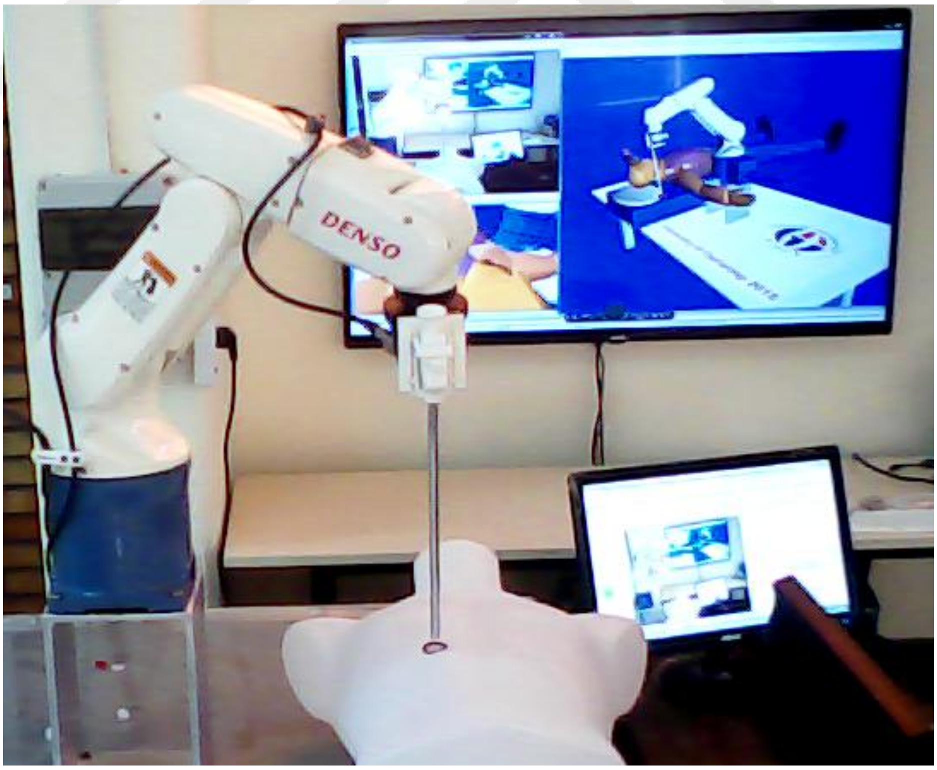


Figure 1.1 Robotic arm medical application

Robotic arm is the branch of robotic technology that deals with the design, construction, operation, and application of an articulated robot. Robotic arm system consists of links of arm, external power source, end effector or tool, controller driving unit, external and internal sensors, computer interface and programs, and input devices as shown in Figure 1.2. The driving unit is a microcomputer with motor drivers to control the motor speed, and motor current (ampere). This unit reads signals from joint sensors, and controls the end effector. Type of motors used in robotic arm are DC motors, DC servo motors, AC servo motors, or step motors. The end effector (gripper) is the device at the end of robotic arm interacting with the environment. The computer is used to control the motion of robotic arm in offline trajectory or online trajectory. Robotic programing skills are required to control robot from computer. The input devices are joysticks, key boards, or haptic ones [3, 4].

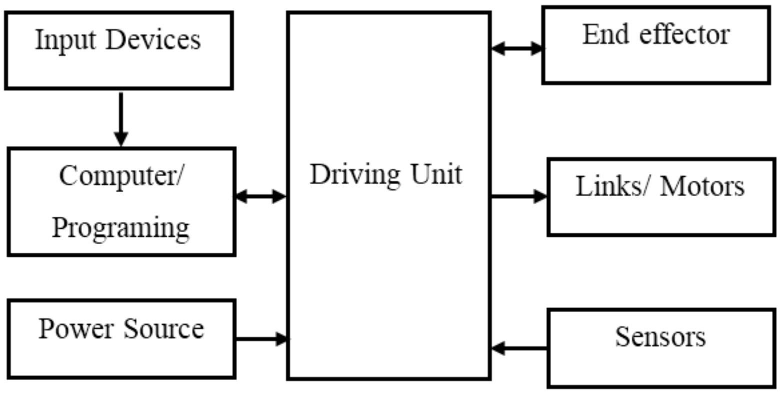


Figure 1.2 Components of robotic system

1.2 Haptic Technology

Haptic technology refers to the study of human touch, and performs force feedback with the external environment. The word haptic is from the Greek "haptesthai," meaning to contact or touch, it means “relating to” or “based on” the sense of touch. Haptic technology is a tactile feedback technology which takes the advantage of the sense of touch by applying force, vibration, or motions to the user [5, 6]. Figure 1.3 shows haptic device to control a robotic arm.

Figure 1.3 Haptic device to control a robotic arm

Haptic technology has been used in this study to sense a touch of applied forces via user interface program with virtual module. The mechanical simulation and virtual module have been combined to assist haptic sensing over a robotic arm [7]. Figure 1.4 shows the haptic system for interfacing between human and machine. Haptic system is basically an inverted robot, it performs to the sense of touch from a computer virtual model, it gives a real world experience from 3D surface model. Haptic devices are enabled by actuators that apply forces to the skin for touching feedback, and controllers. The actuator provides mechanical motion in response to an electrical signal. The haptic system gives the sense for the human operator by the end effector of machine. Human can usually manipulate an object by feeling of hand through the direct contact and utilizing this information to control the motion of object. The machine senses the exerted force and transfers it to simulation with virtual models, the control program applies a necessary torque on robot's actuators [8, 9].

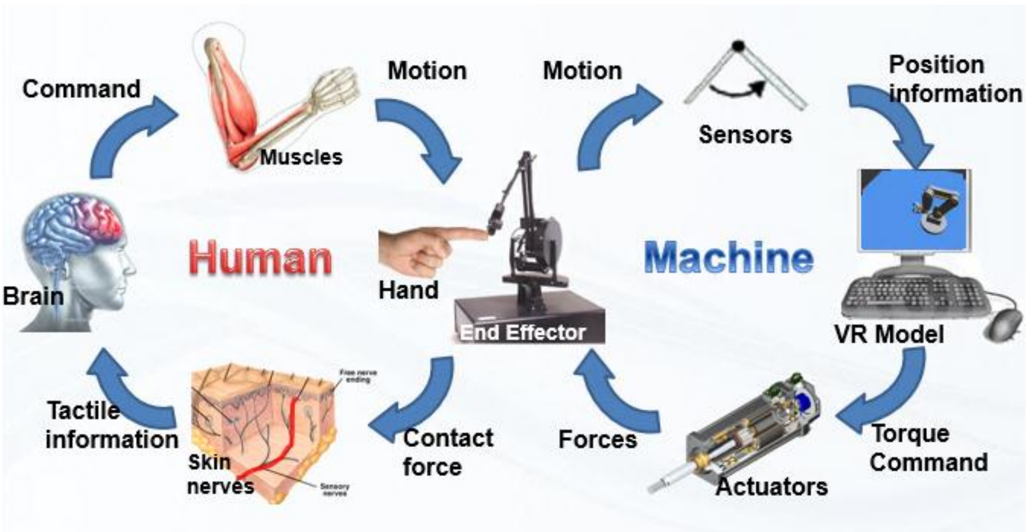


Figure 1.4 Haptic system for interfacing between human and machine

1.3 Motion Control by Human-Robot Interaction

Human-Robot Interaction (HRI) refers to kinesthetic interaction between the operator and the robotic arm. The operator with low experience can reconfigure robotic arm directly by hand. Without need for instruction manuals, the operator can catch the end effector and move it through a desired sequence or path. HRI extends the application of robot. The joint angles and velocities are recorded in control program in teaching time and fed back to robot in time of execution. The advantage of HRI comparing with teach-pendant method is easiness and speed in programming. It can be performed by low experience operator to be associated directly with the task [10-12]. Figure 1.5 shows direct robotic arm control by HRI.

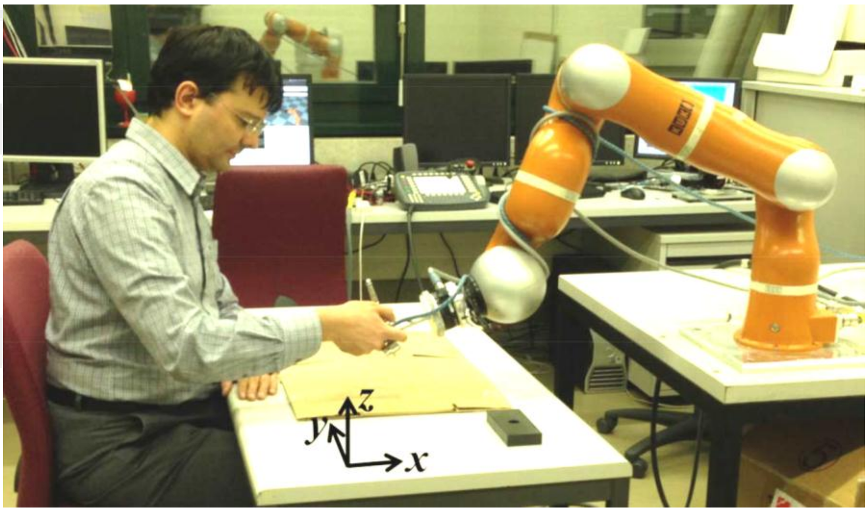


Figure 1.5 Direct robotic arm control by HRI [11]

1.4 Implementation of Artificial Neural Network (ANN) in Robotics

The advantage of neural networks is realized in the facility of implementing in both linear and non-linear systems, so it has capacity to learn system behavior directly from the information available. Linear system models are simply adapted when they derive to system modeling that have different characteristics. The multilayer perceptron (MLP) and back propagation are simple neural network models. A supervised neural network is required in many applications on robotics because it requires a set of input with desired output to learn. The aim of artificial neural network ANN is to find a model that properly maps the input data to the target output data by using historical relations so that the ANN can then be used to find the output when the input is available

[13]. A graphical representation of an ANN is shown in Figure 1.6. In robotics, inverse kinematics has complexity in solution. The task of calculation all of the joint angles for a specific position/orientation (P/O) of an end-effector of a robot arm is proposed by ANN architecture inverse kinematics solution. ANN consists of hidden layers and output layers to solve the inverse kinematics problem for robotics manipulator with 6 degrees of freedom [14].

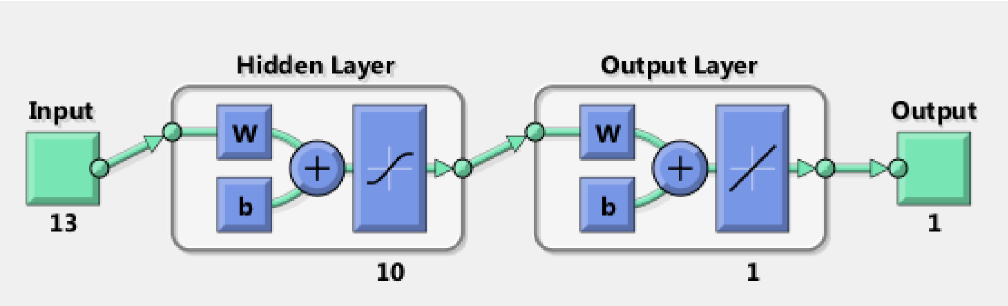


Figure 1.6 Artificial neural network with hidden layers

1.5 Robotic Assisted Minimally Invasive Surgery (RAMIS)

Robotic assisted minimally invasive surgery is the most modern development procedure in minimally invasive surgical techniques. Surgeon is provided with precision instrument using very small point of incision as conventional endoscopic surgery. A main difference is that the robotic arm can offer precise control of surgical tool and visualize the operation area. The robotic assisted procedure allows the doctor to combine both endoscope and regular surgery techniques. The advantages of RAMIS are to reduce blood losses, smaller incision, shortens recovery time, and less exposure of internal organs to external contaminants [15, 16].

The first robotic surgery was implemented in 1997. The surgical robot is developed, and it has been regularly improved. The system basically has got three components: the robotic arm, the surgeon console and the endoscopic stack. The system can control the surgical field and improve the dexterity of the surgeon [17]. Figure 1.7 shows the major technological development related to robotic surgery.

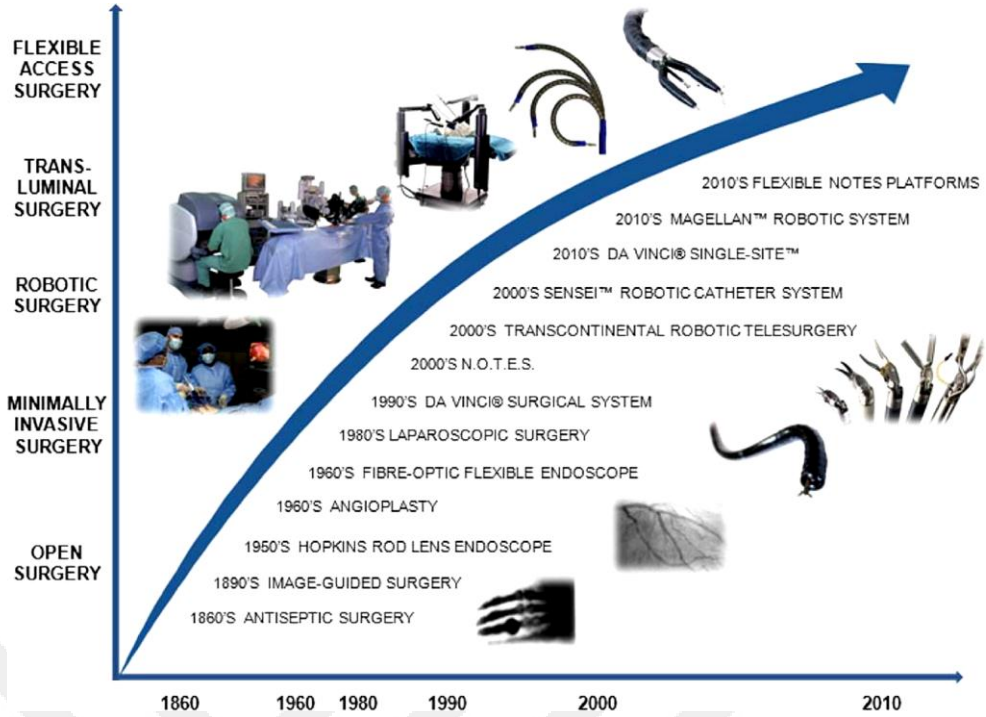


Figure 1.7 The major technology development related to robotic surgery [17]

1.6 Objectives and Contributions

The conventional robotic systems have limited sensing of environment. The force on the end-effector, tool, or environment increase difficulty to perform operations in which the motion presses against an environment. Robot force control has already appeared in a few applications. Robots are capable of performing simple force control that allows them to do some tasks. The expansion in robotic application is in progress towards haptic technology. Robots can do more tasks through the direct physical interaction with human.

The objectives of this study are:

1) Implementation of direct motion learning to a robotic arm based on haptic technology.
2) Designing a Force/Torque (F/T) controller for direct human robot interaction by using six axes F/T sensor.
3) Modulating the kinematics of robotic arm by applying an ANN.
4) Manipulating and controlling of a robotic arm with scenarios of Robotic Assisted Minimally Invasive Surgery (RAMIS).

Many contributions are realized from this study. A new controller of force to position (F/P) and torque to orientation (T/O) is designed. In this controller, external forces are converted to movements, and external torques are converted to rotation. Internal forces and moments from the tool inertia are also eliminated in the proposed controller. System model is designed to get a nominal system condition (zero external force/torque model). The control technique of converting interactive force/torque of human hand to the desired position and orientation of end effector is achieved by developing the virtual spring control method (VSC). Virtual environmental model for robotic system is built by Simulink 3D animation. A real time motion simulation is combined with actual robotic system. Motion learning is performed by different teaching variables; joint position and joint velocity, and force/torque signals.

In this study, a novel inverse kinematics solution is introduced. The method is based on ANN to have an accurate motion control for six axis manipulator robot. ANN architecture has included current joint angles of robot in the input pattern. It improved the performance of proposed ANN while solving inverse kinematics. This architecture is the first ANN that fulfills the requirement of robot precise motion, and reduces joint angle's errors.

Robotic Assisted Thoracoscopic Minimally Invasive Surgery (RATMIS) is also demonstrated by using Denso robot platform. Kinematics of surgical robot mechanism is derived. Surgical manipulations are experimentally executed. A model for endoscope tool and holder part are manufactured for RATMIS.

1.7 Organization of Thesis

This thesis is composed of seven chapters as follows:

Chapter 2 presents a comprehensive literature review including application of haptic technology in robotics, types of control methods, and the advantages of this technique. Artificial neural network in robot motion control, and robotic surgery system mechanism are introduced. At the end of chapter, robotic arm and haptics are included according to the importance and system requirements.

Chapter 3 includes description for robotic system setup, hardware, software, capability and application. Quarc control software, force/torque (F/T) sensor, gripper system, and motion control issues are explained.

In Chapter 4, system modeling and control method are given. Kinematics of robotic arm, forward and inverse kinematics, trajectories, proposed haptic control method, Video-Assisted Thoracoscopic Surgery (VATS), and the proposed control method for RAMIS are presented.

Chapter 5 gives the experimental work. System model verification, the experiments on proposed control method, the online motion learning experiment, and the surgical manipulation scenario by robotic arm are included.

The results are reported in Chapter 6. The experiments on control method are analyzed and discussed. Motion errors are measured and compared with previous studies. Conclusion of the study together with recommendations for future work are presented in Chapter 7. Description of robotic system, and robotic arm data are finally given in Appendix A.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The development of robotics in industrial environments requires a flexible relationship between the machine and the human operator. According to the expansion in scientific application, the robotic arm produces different applications of motion control and task. Robotics assisted minimally invasive surgery (RAMIS) has become one of the most modern applications of robotic technology in medicine. A survey for haptic technology over a robotic arm is introduced in Section 2.2. Then mathematical difficulties in trajectory and path generation are described in Section 2.3. The gain of kinematics analysis solution of a robotic arm is presented, like precise control, and less computing time. Methods and techniques used in RAMIS are explained in Section 2.4. The development in design and performance of the robot assisted surgery is given. In Section 2.5, some remarks of previous studies are stated.

2.2 Haptic Technology in Robotics

Autonomous robotic system needs a new generation of robotic system that is able to achieve different tasks under varying motion conditions by direct sensing of environments. The haptic cooperation between human and robot is controlled according the contact force condition, impedance factors which are adjustable in the system. The haptic system and impedance factors are controlled by an inner motion control loop. The effect of the inner motion control loop against the sensing is reflected in the contact condition. Figure 2.1 shows a haptic manipulation of a mini-excavator combined with a force sensing system. It allows the user to respond quickly to a task environment, changing in time of operation, through the external force reflection in the environmental condition. Robot operations are performed such as joining heavy parts together or press fit task [18].

Figure 2.1 Haptic sensing system combined with a mini-excavator [18]

Haptic interaction can be applied in HRI between the operator and the robotic arm. The operator with low experience can teach robotic arm directly by his hand and move through the desired sequence or path. The industrial tasks of manipulation are much easier to be taught to robot by HRI than commands of programming languages. The physical human interaction with robot is required to teach the robot a specific task. The progress of works in industrial locations requires a tight relationship between human and robot. The progress of scientific application in the field of the autonomous robotic task systems supports the development of robotic systems. It enable robot to accept different motions under changing environmental conditions. The acceptance and the control of robotic systems in industrial environments increase with the degree of independence of robot. Works related sequences and operations are supposed to be taught simply by the online human operator without use of common software design languages but by physical human communication. The particular motion task taught to robot by the human operator has to have F/T or haptic sensors. Figure 2.2 shows a human operator pushes the robotic arm to free position in the workspace. The way to control the system by touch and by the system without any damage is the most important requirement for the described cooperative scenario. The system can have interaction with human and the surrounding objects. The manipulation of the robotic arm can be modified online by the human operator just by moving his artificial body.

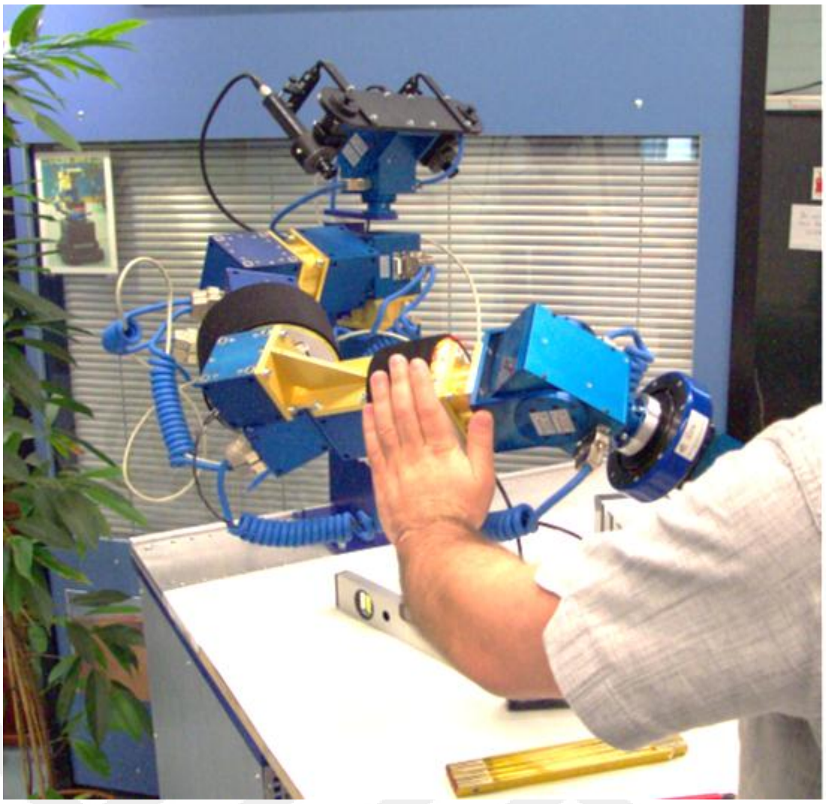


Figure 2.2 Human operator pushes the robotic arm to free position [19]

The cooperation with an artificial system in general is not resulted by the use of sensor. The fundamental structure of robot must have the ability to move to the command input [19]. Direct teaching is a specific process of recording data that represents a motion as a function of time. These data are then applied to robot in a play back time. The motion of repeated path shows slower motion from the recorded data. As there are differences between the real position and recorded data motion. The position error happens during the process. There are several advantages of this technique: very fast robot programming, no need to have a high skills operator, and quick to learn robot motion. The response to automate a task has been progressively growing due to the lack of existing operators. The direct teaching technique for different processes requires playback method. The user teaches the robotic system the motion. The robotic system then plays the recorded path. The control method selects the basic path desired to follow the human operators, the path of task consists of noisy signals, due to the vibration of motion. The paths of robotic arm in HRI process have many types of disturbances such as rigidity of the robotic arm and system response time. Robot with high weight has got low natural frequency [20].

The human teaching and robot playback method need the taught signal to record the task positions of the end effector. In general, teaching articulated serial manipulator is implemented by using a teaching device called a teach-pendant. But teaching with a teach pendant, needs a good operator skill, because there is a transformation between the workspace coordinates and the joint coordinates. The use of a teach pendant is not a basic technique which takes long time. Direct teaching by human hand is movement of robot by a direct applied force. In general, direct teaching is difficult to apply to the robotic arm with joint friction, a high gear ratio, and built-in controllers. Many methods of robotic arm direct teaching is used, such as simulated internal system, hybrid method by impedance control variable, and remote teaching by a joystick. Sometimes these methods are not easy to apply on robot and required tuning parameters, hardware modification, and also gravitational effect. Motion control with free gravity and friction is preferable in direct teaching for a robotic arm. In this case the human can actuate motor directly. This control method is known as '*force free control*'. This lets the robot arm to move by an external force while compensating for gravity and link friction in joint space control. This force free control is an effective method for industrial articulated robot arm in simulation system [21].

The teaching method is categorized into '*off-line*' and '*on-line*' programming according to the connection with robot during teaching period, while the on-line teaching is performed by HRI for the advantages of easiness, and programming speeds. The haptic technology in robotics enables operator with low experience to teach robot completed task. During the process of teaching, the user can grasp robot end effector directly by hand and lead the robot through a desired sequence of trajectory to find path (joint position) and joint velocity of robotic arm. Motion variables are recorded in system control program and replayed during execution time. Because the user can order robot by hand, a sensor is used to get the external force measurement. The suitable signal processing device is F/T sensor. Six DOF robotic arm is upgraded with end effector sensor to have human teaching for specific applications. The haptic teaching has some constraints; especially of sensing range, resolutions, and sensitivity. The control conditions are sequential actions (not loops or subroutines) [22].

A safe motion planning for robot by human guide is demonstrated by using robot sensors and haptic system to improve a control method. 'Virtual spring approach' is used to join a master system and a slave system of robot. The master program for a control force is connected to a haptic unit and force/torque (F/T) signal from sensor. The acquired signal of the forces is then put to control the slave robotic system. Many types of haptic devices are available to develop robotic arm; like Phantom®, Novint Falcon®, and Omega®. Those systems have different payloads, DOFs, sensitivities and mechanical designs. The haptic device does not have same workspace as a robotic system. The size ratio between workspace of a haptic device, and robotic system workspace is generally lower than 1/10. Low size workspace is not applicable for robotic systems that need a very accurate motion. Large size workspace with haptic unit is useful to have high range of motion for robot. When robot arm is tele-operated by a haptic system, it is able to manipulate it in a complex 3D workspace with an obstacle, precise motion control, and avoidance of collision to reach goal position [23].

A haptic system is used to have interface with a system in virtual space. The force signal is controlled over the visual system that makes the design more approachable and precise. There are two general haptic control methods usually applied for the control of haptic systems. First control method is 'impedance design of controller' in which the position is the input variable, and the feedback force is the output to the user. The second control method is 'admittance design of controller', in which the force is applied by operator as input variable, and the position is the output to system. Both controllers, impedance control and admittance control are used for cooperating between a virtual system and a haptic system. Figure 2.3 shows block diagram of haptic interface with a virtual environment. A work was implemented to estimate the stability of impedance control method in haptic system interaction design. The application of impedance control method in a robotic arm with haptic interaction is very wide. Closed loop control and open loop control of impedance control method are applied for general purpose robotic arm. Many control variables are studied for 2 DOF robotic arm planar motion system. The visual force feedback is applicable by using a haptic system [24].

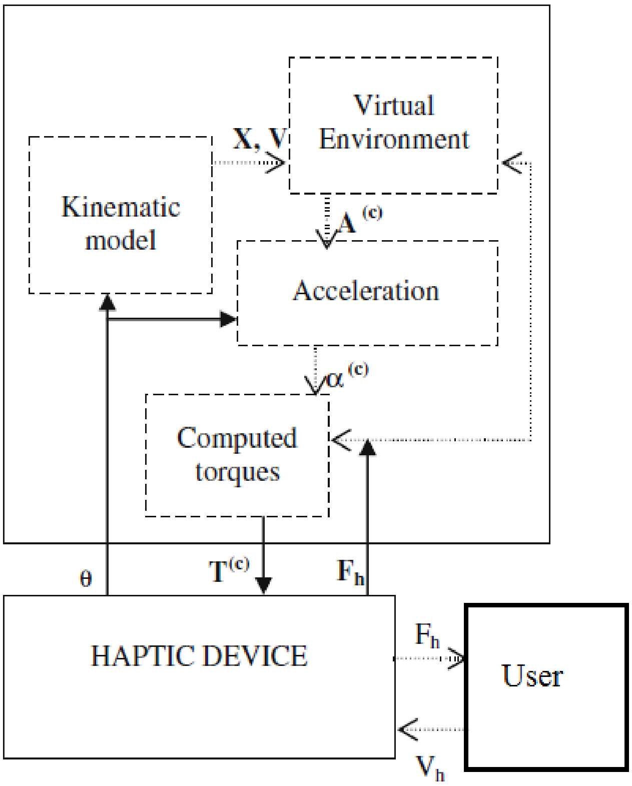


Figure 2.3 Block diagram of haptic interface with a virtual environment [24]

A 5 DOF haptic system was modified to 7 DOF system as shown in Figure 2.4 to provide force feedbacks to surgeon during robotic surgery. A sensor is used over an endoscopic tool to have a haptic system. A control method was applied for main robot component to control a surgical tool. The force feedbacks from slave system were reflected on master control system [25, 26].

The behavior of robot is described by the steady state of joint position, the simple controller is PID type. Velocity control error is depended on the gain values. It is combined with the robot position error to have system trajectory errors and command control system. The robot joint position and velocity error during execution of motion is the most difficult problem, as a result of error. Several control methods are applied for robotic arm to obtain a reduction in tracking error. An approximation free controller on KUKA LWR4 robotic arm is applied [27]. Rolling of fingertips gives precise motion of interaction positions in dynamics design of grasping. The friction to the contact point in robot of hands increases the required force on control design.

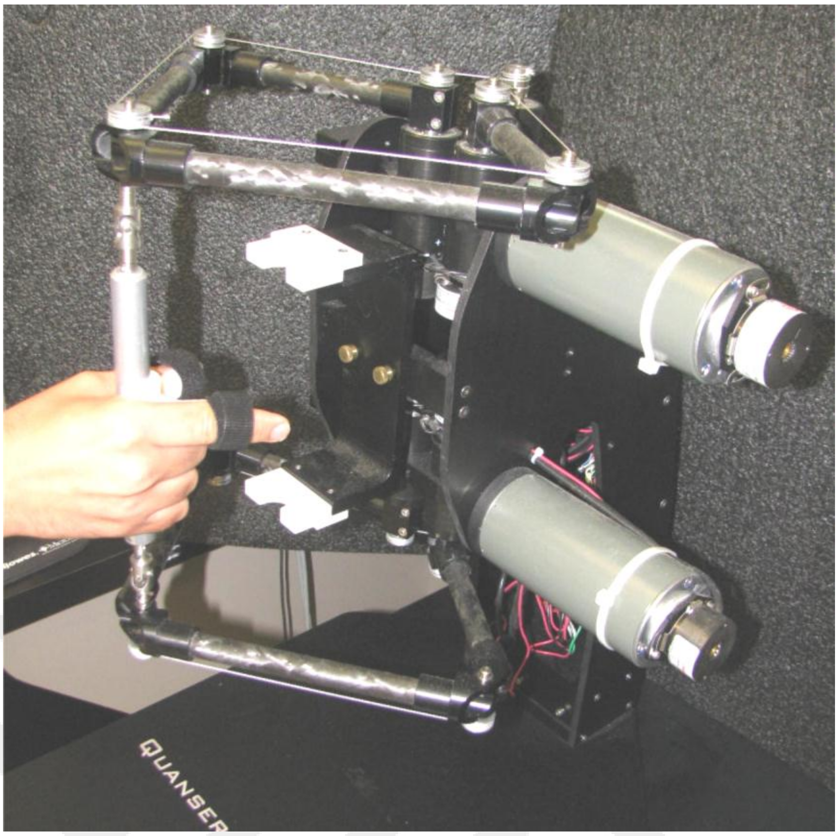


Figure 2.4 Five DOF haptic system modified to a seven DOF device [25]

The equation of motion for this robot includes the rolling constraints for most researches. The friction may not be enough to stand the tangential component of the applied force. Human hand was coupled with the fingertip for motion of rolling. Force trajectory generation with position control was applied. The control design was the transient and steady state motion of motion behavior. A simulation was designed for 5 DOF robot to perform rolling contact in case of friction state [28].

The development in robotic arms are implemented by using double robotic arms. To achieve some application in robotics, a hydraulic robotic system is used for high payload and power. Many researchers have performed the dynamics and control design for this type of robot. A virtual spring-damper (VSD) control method is also applied on end effector of robotic arm to control the position. Figure 2.5 shows VSD control method and robotic arm model used in the simulation. The control method based on the force and desired position was studied without considering to the tool inertia. Therefore the result shows an error along the axes of motion. The position error is reduced by using dual VSD controller [29] as shown in Figure 2.6.

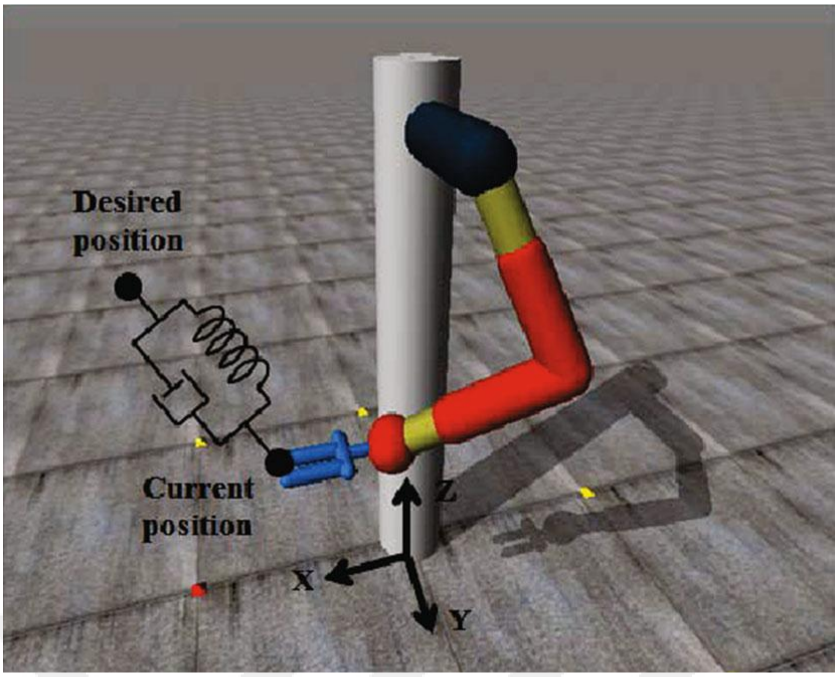


Figure 2.5 VSD control method and robot arm model in the simulation [29]

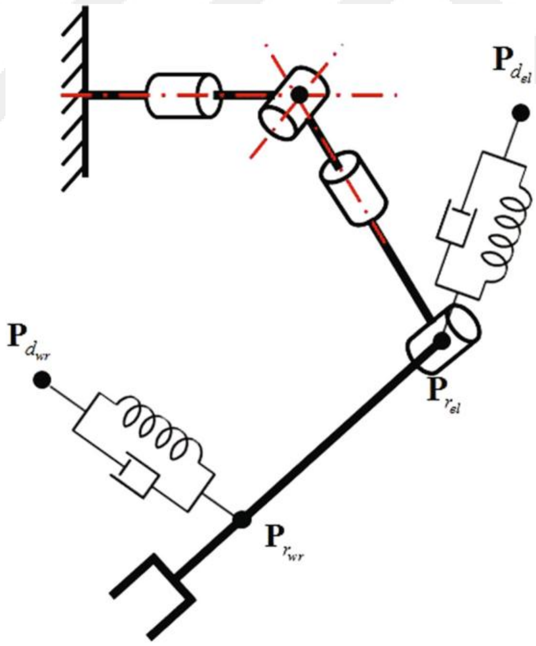


Figure 2.6 Dual VSD control method for robotic arm [29]

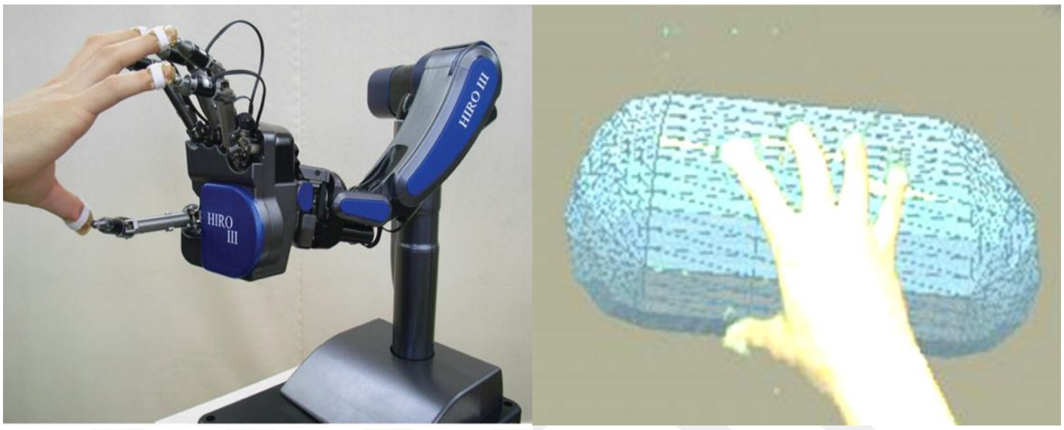
Haptic system can produce an interface between real environment and virtual reality environment. The contact force interacts with virtual object, the motion of haptic device by user is mapped to simulation world. Collision avoidance and force sensing are responded in computer model haptic interaction. The design of haptic system and solving singularity problem are the most complex part, the trajectory and motion control algorithm are performed in many studies. The physical constraints like friction and inertia arm are difficult in control methods. The tool inertia effect on stiffness perception of simulated virtual objects is realized. A holding pivot point on the end effector, and estimating the acceleration of the control point are used to compensate the inertia effect [30].

The skill training from human to robot becomes the most important work in area of haptic robot for industrial and medical applications. Sometimes it is difficult to train medical students with real human organic part or using animals for training. So training with virtual medical models have expert of contact force and position information which are simple to be delivered by teacher. Applying skill on virtual reality needs a haptic system interface. Training system with force and touch sensing gives the model more reality. Figure 2.7 shows skill training from human to virtual system. The method of record and replay can be used to transfer force/position data to the virtual model. The force/position is transferred from fingertip to 3D space system by using record and play teaching strategy. The skill translation between user and virtual environment is achieved [31].

The teaching of exerted forces and the movement of the bio-manipulation can be transferred to virtual display system. The sensing of applied forces for an object in a virtual environment to create a haptic interaction system is explained. Figure 2.8 shows teaching force from a bio-manipulation system. Position accuracy is important for goal object position manipulation.

Human friendly system is based on master and slave multi systems. To improve position accuracy and force sensing resolution, a virtual reality simulation model was designed; like flexible multi-function bio-manipulator system. Robotic assisted cell injection system is developed by using haptic interface. Manual Intra-Cytoplasmic Sperm Injection (ICSI) requires long time training but still has low success rates primarily because of poor control over the injection applied force. Automated cell injection system with visual tracking of cell and injection force characterization are inevitable. Human friendly bio-manipulation and tweezing assistance robot are very necessary for the conventional instrument's difficulty. Low success rate is caused by several difficulties in treatment bio-organisms. Slippery surface requires external process to handle it. Deformable tissue needs force to be controlled accurately [32].

The robot can be interacted with user by speech commands, physical interactions, and vision orders. Human robot interaction combines these actions to achieve task learning. Human computer interface is the area for human robot interaction, the number of inputs variables can be reduced by direct user cooperation. The robot learning controller is based on sensor information as well as the human actions in virtual tracking human environment [33].



(a) (b)

Figure 2.7 Skill training from human to virtual system (a) Finger skill transfer system, (b) The virtual reality skill recording system [31]

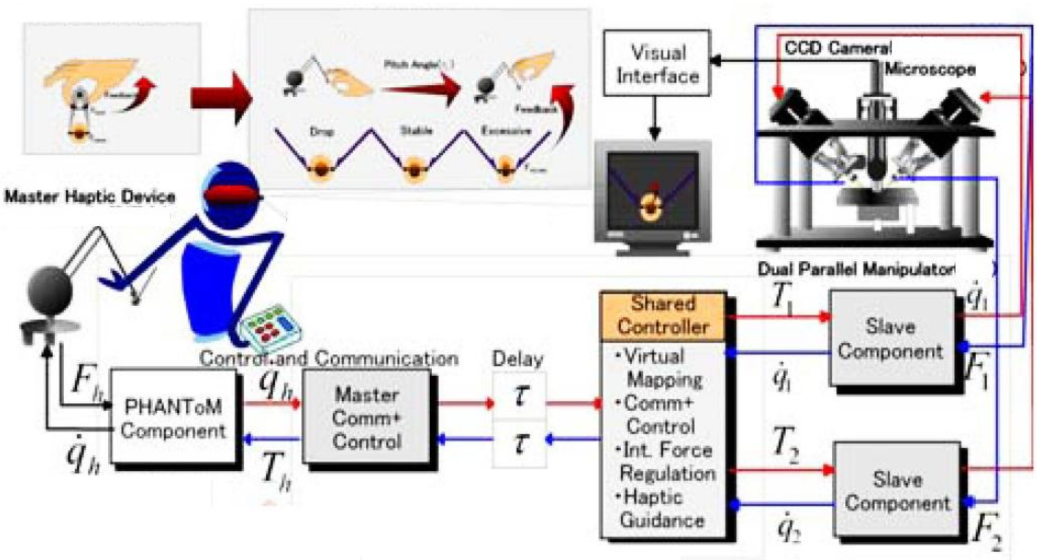


Figure 2.8 Teaching force for bio-manipulation system [32]

2.3 Artificial Neural Network (ANN) in Robot Motion Control

Artificial Neural Network has become the most modern technology of robotic motion control. It has many gains in robotics like a precise control, less computing time, and solving some mathematical difficulties in trajectory and path generation. The solution of robotic arm kinematics is important in online motion, solution of inverse kinematics is mathematically difficult and requires high time process. Most tasks of manipulation in robotics need Cartesian coordinate in movement. In inverse kinematics, chosen position and orientation of the tool in Cartesian coordinate are available. The joint positions in the joint coordinate are found. The general solution of inverse kinematics for robotic arm are geometric, iterative, analytic, or algebraic approaches. Recently, artificial neural network (ANN) methods for inverse kinematics are used. Many researches are applied on ANN for robotic arm to solve the singular problem. A 3 DOF robotic arm was used to find the inverse kinematics by inversion of multilayer network method, the joint position were predictable for desired end effector position. The solution was for simulation of three links system. The findings have shown an estimation for the joint angles. The work was based on examining the implementation of neural network with multiple joints robot. Figure 2.9 shows ill-position solution in inverse kinematics. The motion of robot is carried out in two dimensions [34].

Singularities and uncertainties in robotic arm joint configurations are complex points in the inverse kinematics of robot motion control. A realistic solution is obtained by using heuristic methods. Neural network is designed for a general robot. Figure 2.10 shows ANN structure. The goal of this research was the use of ANN to teach the robot features rather than having to identify an obvious robot system model [35].

A neural network (NN) and genetic algorithms (GAs) were applied to find the inverse kinematics solution of the robotic arm. The objective was minimization of the error at the end effector and enhancement for the joint position of the inverse kinematics problem [36].

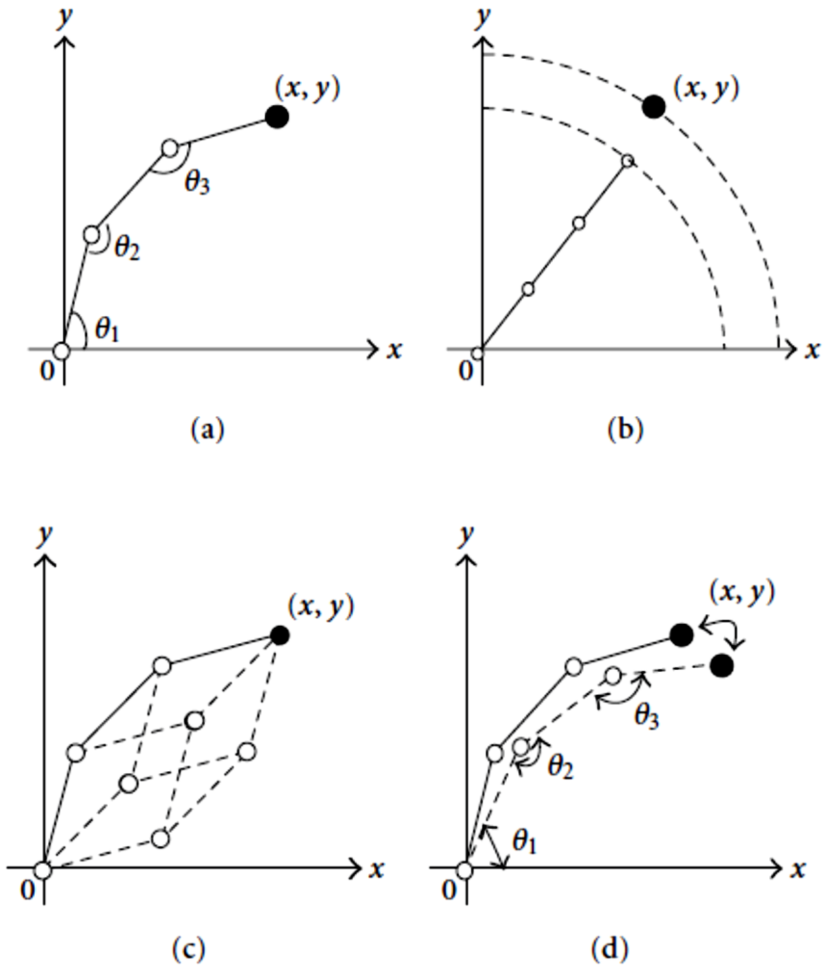


Figure 2.9 Inverse kinematics problem of robotic arm, (a) general joint angles solution, ill position in terms of (b) outside of existence, (c) uniqueness solution, and (d) error and stability [34]

The inverse kinematic of redundant manipulators is suggested by using Neural Networks (NNs). The joint positions of robotic arm are controlled in real time using the Cartesian coordinate of the end-effector. Position errors of end-effector and stability of the joint angles are achieved as shown in Figure 2.11. An artificial neural network was implemented for 3 DOF robotic manipulator [37]. The introduced method is a non-linear relation between Cartesian and joint coordinates by using multiple layer perceptron in ANN. Simulation executions were applied. The inverse kinematic approach is used to solve redundancy by two methods, gradient projection and task on space increasing. The closed loop of Cartesian error was designed for joint movements, to avoid obstacles and have improvement in some drawbacks. A damped least squares method with Genetic Algorithm (GA) was applied for smoothness of motion [38].

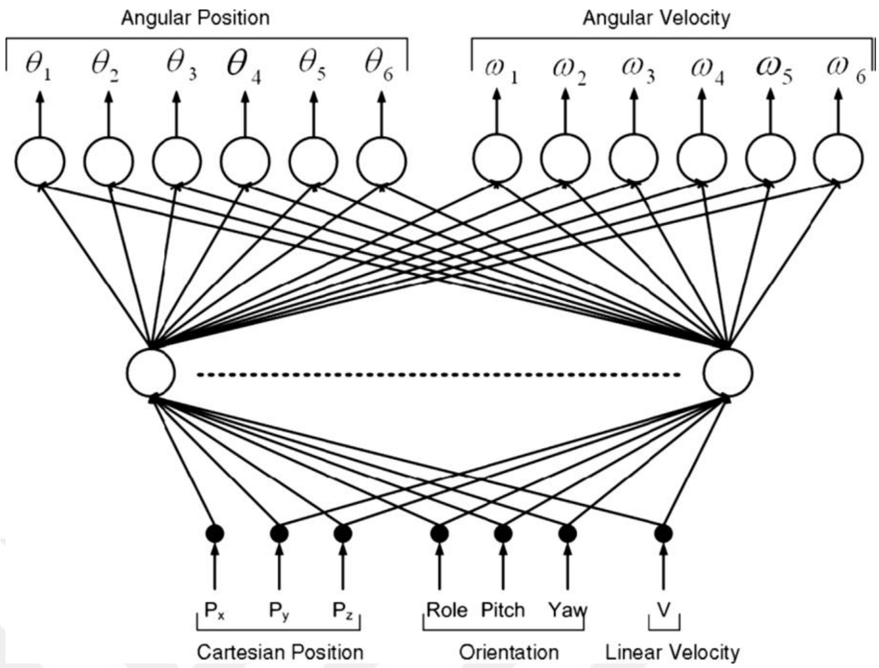


Figure 2.10 ANN designed for robotic inverse kinematics [35]

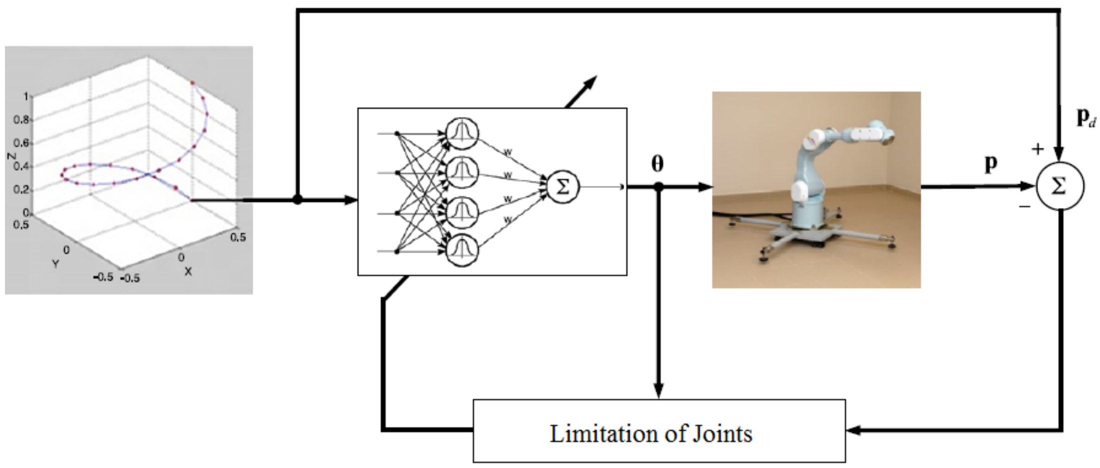


Figure 2.11 ANN implementation for 3 DOF robotic manipulator [37]

A neural network architecture is designed to solve the inverse kinematics problem for robotic arm with 2 degrees of freedom. The ANN is multi-layered perceptron (MLP) with a back-propagation training algorithm for tuning the difficulty of the algorithm. Inverse geometric of robotic arm is used for calculation of matrix inversion [39].

A high-order-logic (HOL) approach for solving the kinematics analysis of six axis revolute joint robot is introduced. The formula was necessary for massive amount of user interference to overcome the restrictions of kinematic analysis. There were two important points for the kinematic analysis in HOL: formal geometrical principles and libraries of screw rigid body theory, which are important tool in analyzing the robot. Figure 2.12 shows the formula of high-order-logic procedure in solving inverse kinematics of 6R robotic arm [40]. The inverse kinematics problem of the 6-DOF robot was solved by using curved-surface scanning with ultrasonic trying task. The method of the shortest distance was applied to solve the inverse problem of robotic system. A three dimension application software was used to simulate manipulation of ultrasonic trajectory and path generation. The method was optimized to have only eight solutions in trajectory control. It was based on robotic arm sequence joint axis interest and shortest distance instructions to adopt optimal solution. The simulation trajectory was formed for complex shape motion [41]. The kinematics and singularities of an asymmetrical parallel robotic wrist were studied by using the method of Lagrange multipliers and considering all mobile components. The designed model was numerically demonstrated to show its computation of precision [42].

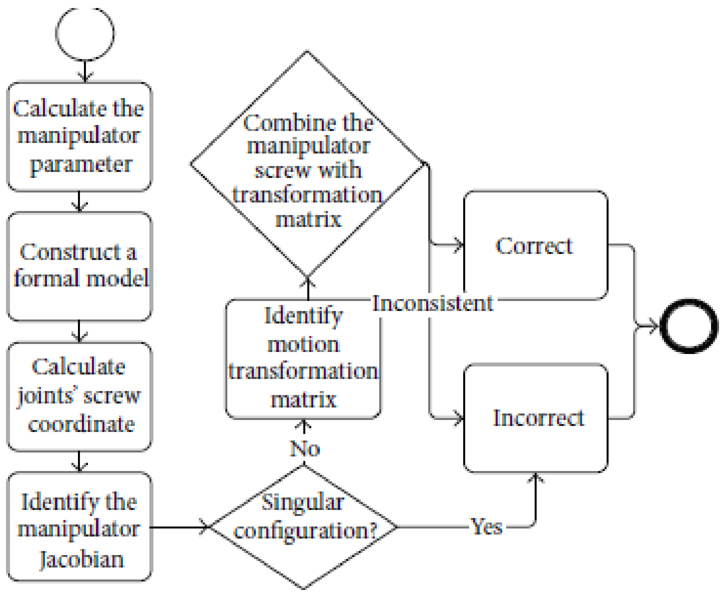


Figure 2.12 Formula of high-order-logic procedure in solving inverse kinematics of 6R robotic arm [40]

A method for solving the inverse kinematics problem by using ANN was presented for a PUMA 560 robotic arm. An inverse kinematic solution is applied by training the neural network with positions of the robot's end-effector in Cartesian coordinates and its corresponding joint angles. A 3D environmental model is presented for the robotic arm by developing a model in MATLAB. The method showed mean square error (MSE) 1.2178. The regression value is obtained 0.87527, and the position errors are found in x, y, z axes 4.93%, 7.29%, 3.73% respectively [43]. A solution of the inverse kinematics was applied for generating suggested trajectories in two dimensional Cartesian motion. A feed-forward neural network was used for planar robotic arm of three-link manipulators. A circular path was performed by method in Figure 2.13. The result showed the best performance at epoch 9 with mean squared error MSE of 0.0054387 [44]. The kinematics of three DOF was introduced for the leg of the humanoid robot as shown in Figure 2.13. Decoupled closed form solution for the position and orientation was the kinematics analysis, the joint orders were presented by Denavit Hartenberg (DH) transformation matrices. Swing phase equations were established to avoid matrix inversion problems. The approach of finding joint position form geometry was based on triangular preparation for active sequence of joint angles in DH chain. The orientation function was obtained for matrix series in a straight forward configuration. The final orientation is allowed to be parallel to ground [45].

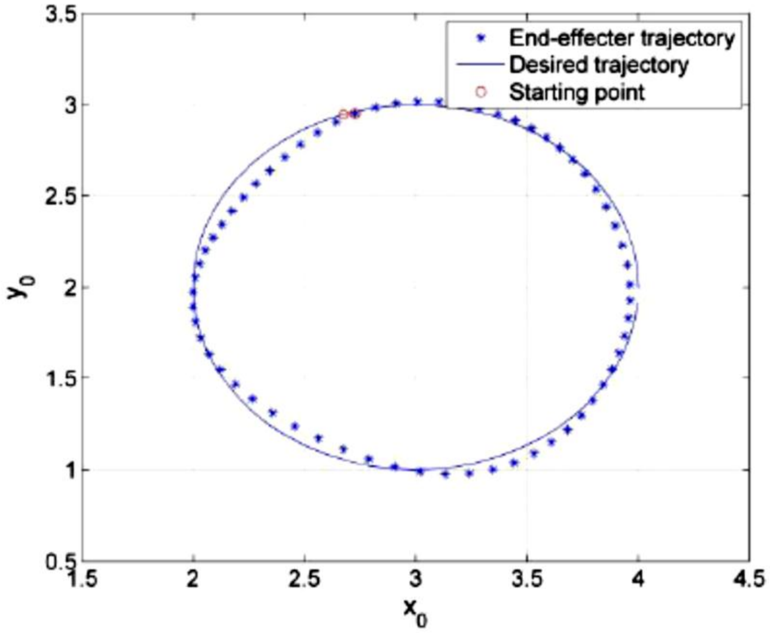


Figure 2.13 Performing circular path by planar robotic arm [45]

The kinematic analysis on industrial robotic arm were disturbed by vibrations effects. The error in trajectory was enhanced by sensing with an accelerometer and a gyroscope. The path profile was configured for the joint. The position tracking of welding task was estimated [46]. Robot kinematics was identified by geometry kinematics approach to map human arm joint positions. The stiffness index was controlled by hand gesture. The stiffness of human arm was estimated within robot workspace by experimental stability task. A moving task was applied to test the behavior of geometry kinematics approach on robotic simulator [47]. The geometric approach was used to find kinematics of a robotic arm. This modeling and analysis approach was executed by using a five DOF arm with a gripper mounted to the *iRobot* mobile platform [48]. Real time robot kinematics factors errors estimation based on inertial calculation unit are presented. The orientation of robotic arm with the orientation of the method in online are obtained. This method is combined with Factored Quaternion Algorithm (FQA) and Kalman Filter (KF) to find orientation position of end effector [49]. A 5 DOF robot is used to find an analytical solution of inverse kinematics in spatial parallel micro-manipulator. A structural model of system geometry are presented for the micro-robot's task [50]. Inverse kinematics and direct kinematics were analyzed and simulation is performed for joint angles and parameters of link for six axes robotic arm. Trajectory generation is introduced for the applied path motion of the manipulator as a time sequence task [51, 52].

2.4 Robotic Surgery (RS)

Robotics assisted minimally invasive surgery (RAMIS) has become one of the most modern applications of robotic technology in medicine. It has many advantages in performance such as precise control, less pain, trauma, blood loss, scarring, in addition to low recovery time but some drawbacks are available like constrained degree of freedom (DOF), lack of force sensing and restricted vision. Based upon the motion of the surgical instrument, to execute Minimally Invasive Surgery (MIS) task inside the patient's body can be classified as either hand or computer controlled. Based upon the type of operation can be classified as an active, passive or co-manipulated RAMIS near the patient. Remote center of motion (RCM) mechanism is usually used to control the surgical tools, it has a fixed incision point (trocar), either mechanically or virtually associated with the mechanism, the surgical tools are inserted and rotated through this point. In general, this point has motion with four degrees of freedom and no lateral motion around this point in order to avoid damaging the patient's skin, while the tooltip has six degree of freedom.

Many researchers are focused on advantages and assistances of robotic arm in surgical operations. They are developed tools for assisting medical task. A wide area of study has been done on the improvement of robotic arm and surgical tools for robotic assisted minimally invasive surgery (RAMIS). It is not an easy procedure. A smooth and accurate motion is required. It sometimes causes fatigue, pain, and poor ergonomic positions. A studies were done on the implementation of robot in MIS. Some of them designed surgical robot or forceps for MIS, and others suggested solutions for the problem of absence in precise motion and kinematics of robotic surgery.

Haptic system in RAMIS has many limitations, because the robotic arm contacts tissues in all sides around trocar point, it is difficult to find the position of contact force applied on the endoscope. Some studies are focused on the contact sensation on surgical grasping tool. The robotic system is usually equipped with force/torque F/T sensor to measure the applied forces on the endoscope. The robot can be developed to provide a force sensing for medical applications in minimally invasive surgery (MIS). The force of pulling or grasping the tissue is sensed by surgeon hand and safer tele-operation control can be achieved [26].

Depending on RCM at the insertion point, a surgical manipulator was designed. The manipulator works in MIS with virtual remote centered motion, which is the distal rolling joint. The performance of the designed manipulator was satisfactory for simulated MIS. The position error was effected by inertia of surgical manipulator arm. The motion of this arm produces moment of inertia over joints and base which change the motion tracking performance. Figure 2.14 shows the designed surgical manipulator [53].

Mechanisms for surgical robot was developed for MIS assistant robot, kinematic analysis was applied for the manipulator and investigated in different configurations. Measured values are combined with the standard deviation to achieve optimum results with set of constraints at task manipulation. The optimization technique was Global Condition Index (GCI). A prototype was developed as shown in Figure 2.15 based on the required motion and RCM mechanism [54].

Figure 2.14 Designed surgical manipulator for MIS [53]

A parallel robot was designed with RCM to have a better rigidity and accuracy, the inverse kinematics were analyzed and global performance index (GPI) was used to optimize the structure design. In RAMIS, the main motions of doctors can be imitated by a robotic arm which are attached to a surgical tool through kind of free motion in Remote Center of Motion (RCM). The motion has two dimensional rotation around a fulcrum point, one dimensional translation along the instrument axial direction, and one dimensional rotation around as shown in Figure 2.16. The 3D model of the parallel robot including joint space configuration is given [55].

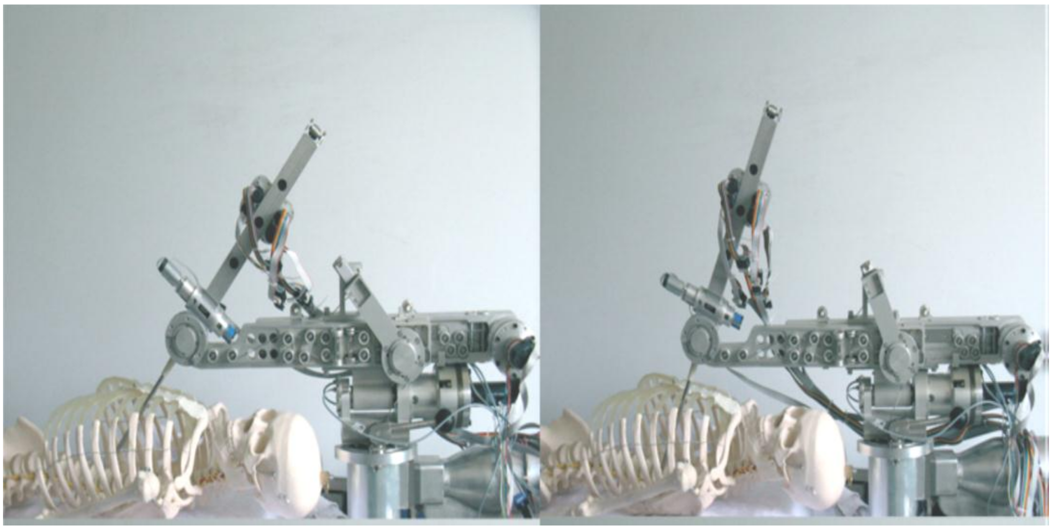


Figure 2.15 Prototype for the MIS robot [54]

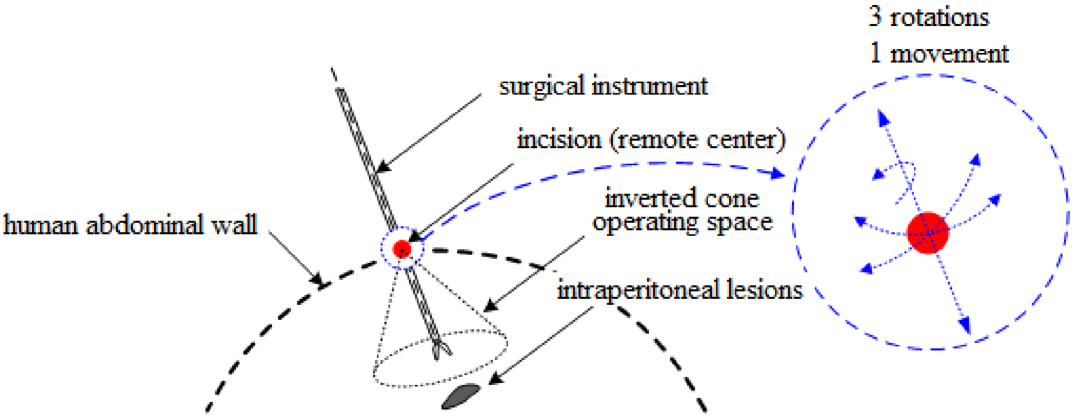


Figure 2.16 Free motion in remote center of motion (RCM) [55]

An approach of programming RCM was implemented in order to have robot manipulation under constraints of motion and insertion. A parallel robot was used to assist [56]. A spherical mechanism was designed with a revolute joint and a curved sliding joint for robotic needle insertion. It moves on the surface of a sphere, the mechanism was based on the RCM to avoid contact with the patient's body. An analytical solution for the direct and reverse kinematics of a mechanism was used in a robotic needle intervention system. Figure 2.17 shows the spherical mechanism for robotic needle insertion. The system is demonstrated in workspace and achieved the desired motion at the tip of needle [57].

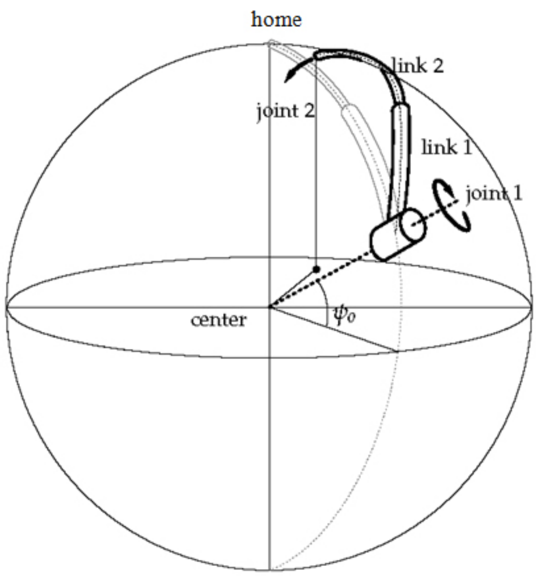


Figure 2.17 Spherical mechanism of robotic needle insertion [57]

A virtual surgical model can be used in exercising on RAMIS. A characterization of RCM constraint was given to achieve a virtual task control and satisfy MIS requirements. The mechanical lacks in RCM are reflected from the control constraints of manipulator. Programmable RCM for serial robotic arm allows the system to be used in open surgery and minimally invasive surgery. The passive joint control was applied in real time optimization method. Figure 2.18 shows execution of virtual surgical insertion task [58].

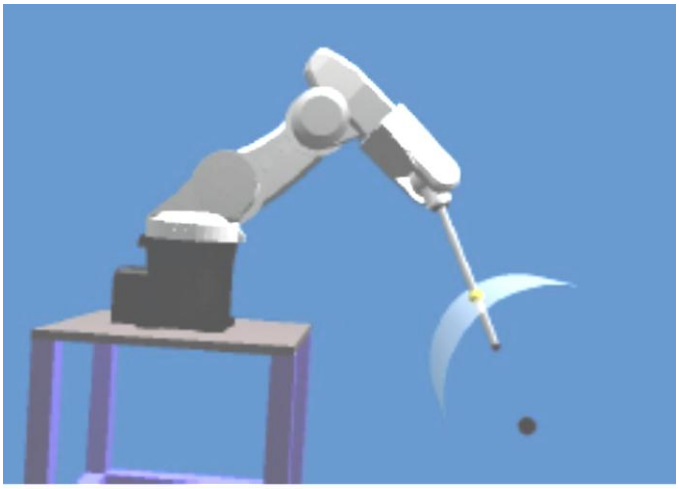


Figure 2.18 Virtual task execution to control on the insertion [58]

A design and optimization of remote center of motion mechanism of minimally invasive surgical robotics were studied. MIS has got a limited view is surgical area and difficult of control surgical parts. The RCM mechanism was developed with high stiffness structure and gravity compensation for surgical tool [59].

Mitsubishi PA10-7C robotic arm (7 DOF) was used for RAMIS. The manipulator was adapted to 4 DOF endoscopic surgical tools. Mechanically constrained kinematic structures has got a physical structure that pivots about the desired RCM. Mitsubishi robot provides high rigidity and the controller faults cannot cause disturbance in the RCM constraint. Figure 2.19 shows Mitsubishi PA10-7C robotic arm performing RCM. Experimental test showed end effector error for the motion of RCM, the maximum end effector error value was average 6mm for actual trajectories of different RCM locations [60]. A surgical robotic system, including the cone type remote center of motion (RCM) mechanism and two articulated instruments to have a flexible linkage driven elbow was introduced to single-incision laparoscopic surgery operations. [61]. In kinematics of motion control, the problem of singularity has got a high majority. A control architecture in MIS was suggested by introducing a Jacobian matrix for RCM for both end effector and trocar point. It showed a reduction in error of motion through the movement in singularity configuration [62]. A kinematics and workspace modeling of a parallel hybrid robot was designed for minimally invasive surgery as shown in Figure 2.20 [63].

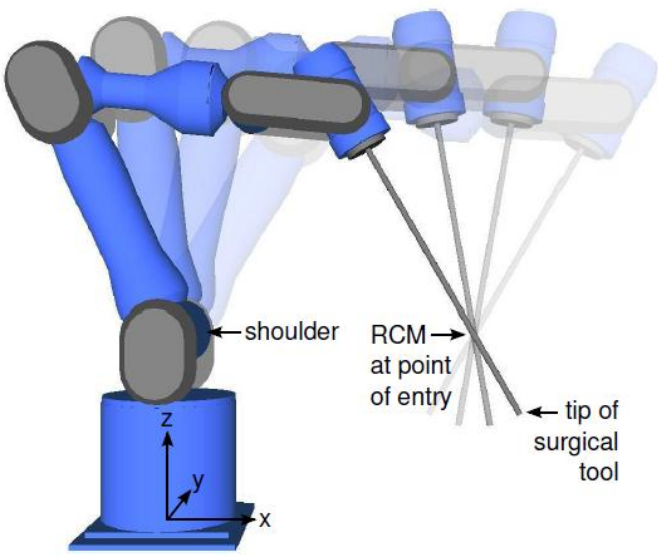


Figure 2.19 Mitsubishi PA10-7C robotic arm performing RCM [60]

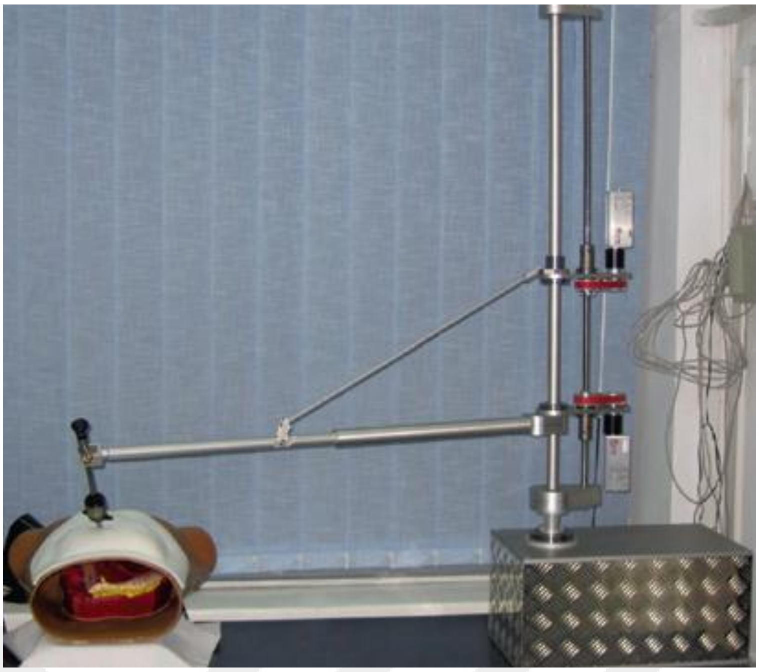


Figure 2.20 Parallel hybrid robot used in minimally invasive surgery [63]

A model based design analysis for a programmable remote center of motion in minimally invasive surgery virtual model is implemented based on the derived mathematical models of Simulink graphical programming [64]. A compact and accurate positioning mechanism, the hybrid mechanism based on a parallel linkage was used to provide three rotations and one translational motion for single port actions. For remote center of motion, the solution of the inverse and direct kinematics was developed [65].

2.5 Remarks on Previous Studies

The previous studies in haptic technology with robotic arm are illustrated that force/position (F/P) controller is a good way to teach the robot different tasks under the environmental changed or human interactions. In HRI, the robotic arm becomes free for movement by any external force and torque. Most studies was to design a force to position controller for end effector by using HRI while the orientation of end effector controlled by external torque in Cartesian coordinate, the torque control were considered in many researches under the joint coordinate. The tool inertia force compensation was included in the design while the moment of inertia had an effect, and it should be also considered. A Different control technique of converting the interactive force of human hand to the desired position of end effector was used to eliminate position error of robotic arm.

Previous studies on robot kinematics have introduced solving methods for six axis manipulator robot based on different approaches in motion control. It is clear from previous survey that no study has included current joint angles of robot in ANN. The ANN architecture has been used to improve the performance of robotic system in solving inverse kinematics. It was focused on fulfilling the requirement of robot precise motion and reducing joint angles error for robot tasks. The robots were used in wide range of tasks increasingly in medical appliances. Many robotic arms were designed and introduced by solving method for RCM kinematics used in RAMIS. The robot architecture increases the diversity of robot in MIS. The researches tried to explain the requirement of surgical robot and how it can improve and enhance performance and outcomes in some aspects of the robot assisted surgery.

CHAPTER 3

ROBOTIC SYSTEM; HARDWARE AND SOFTWARE

3.1 Description of the System

The robotic system is an open source architecture robot from Quanser robotics research technology. The robot is a six axes Denso robot arm. This platform is an industrial robot. The system is a workstation for advancing research and teaching of robotics, the range of applications for this system are robot tele-operation, pick and place task actions, human rehabilitation system, and robotic assisted surgery [66]. Figure 3.1 shows Quanser open source architecture system.

The acceleration and controller design of the system increases the implementation area and research capability of system. This system is appropriate for operation in real time motion control. The software of system is Quarc control program which is used to control different mechatronic systems by an interactive MATLAB® / Simulink®, C/C++, C.NET, or LabVIEW. The development environments of Quarc are perfect for design, simulation, implementation and experiments of interface system communications, control, signal processing, and standalone controller execution.

Figure 3.1 Quanser open source architecture system

The Pre-Configured PC is a dedicated computer to setup the workstation and execute programs. The system includes the following hardware and software:

i) Denso 6-axis articulated robot.
ii) Quarc control software.
iii) Pre-configured PC.
iv) Driving Unit.
v) Force/Torque sensor (F/T).
vi) Gripper system.

All these parts are installed and tested in Mechatronics Lab., Mechanical Engineering Department, Gaziantep University.

3.2 Denso Robotic Arm

Denso robotic arm is a 6 DOF serial manipulator. The robotic arm has six motors and encoders to measure the angular position of the motors. The payload is 2 kg. The maximum compound speed is 3.9 m/s (at the center of end-effector mounting face). Denso robot links and joints configuration are shown in Figure 3.2. The maximum allowable moment of inertia over joint 4 and joint 5 is 0.3 kgm^2 and for joint 6 is 0.007 kgm^2. The repeatability in each of x, y and z directions equal to ±0.02 mm. Denso Robot weight is approximately 14 kg. It can be mounted to floor, and ceiling positions. The overall arm length is 420 mm.

The actuators are AC servomotors with brakes for all axes. The position detectors are absolute encoders. The encoders and motors specifications are summarized in Table 3.1. This table lists the encoder's resolution, motor's torque constants, and joints hard stop limits. The system contains six amplifiers operating about each motor with built-in feed-forward and proportional-integral-derivative (PID) controllers. The rate of processing for each controller is 1 kHz. The controller of system is Ethernet based computer communications. All the gains in the built-in controller are accessible from Quarc blockset. These blocks also have real-time access to the amplifiers current values in the system. The built-in controller gains can be tuned in Quarc interface, or a new controller design can be built in MATLAB/Simulink environment with fully open-architecture operation. Appendix A describes the mechanical specification and workable space of the robot.

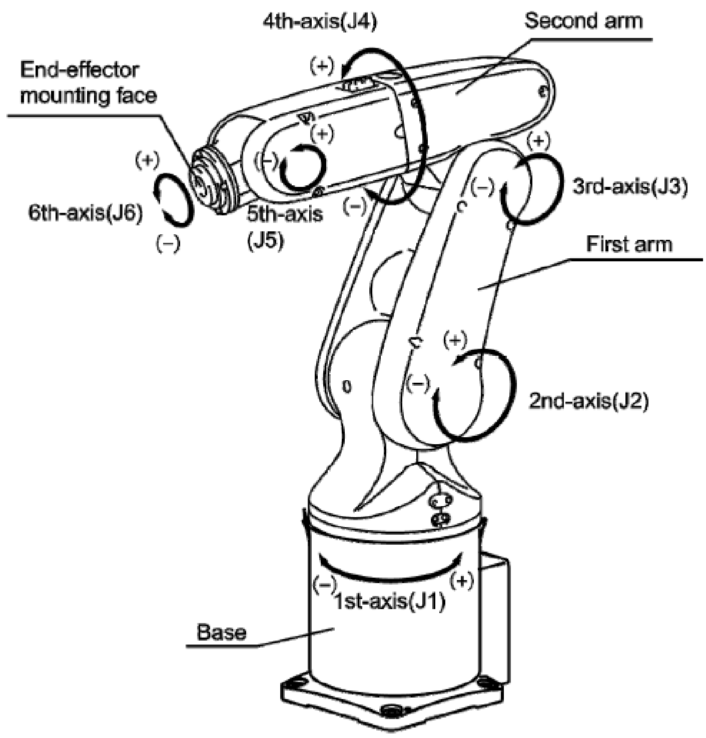


Figure 3.2 Denso robot links and joints configuration [66]

Table 3.1 Motor and encoder specifications for the robot.

Joint No.	Encoder (Count/deg.)	Torq. const. (N.m/Amp.)	Joint Max. limit (deg.)	Joint Min. Limit (deg.)
1	43690	0.38	160	-160
2	58254	0.38	120	-120
3	43690	0.22	160	20
4	36408	0.21	160	-160
5	36408	0.21	120	-120
6	36408	0.21	360	-360

3.3 Quarc Control Software

Quarc is an open architecture control module, it has all the capabilities of interfacing with Simulink library. It is a collection of blockset programmed for the robot system. Figure 3.3 shows Quarc control software structure.

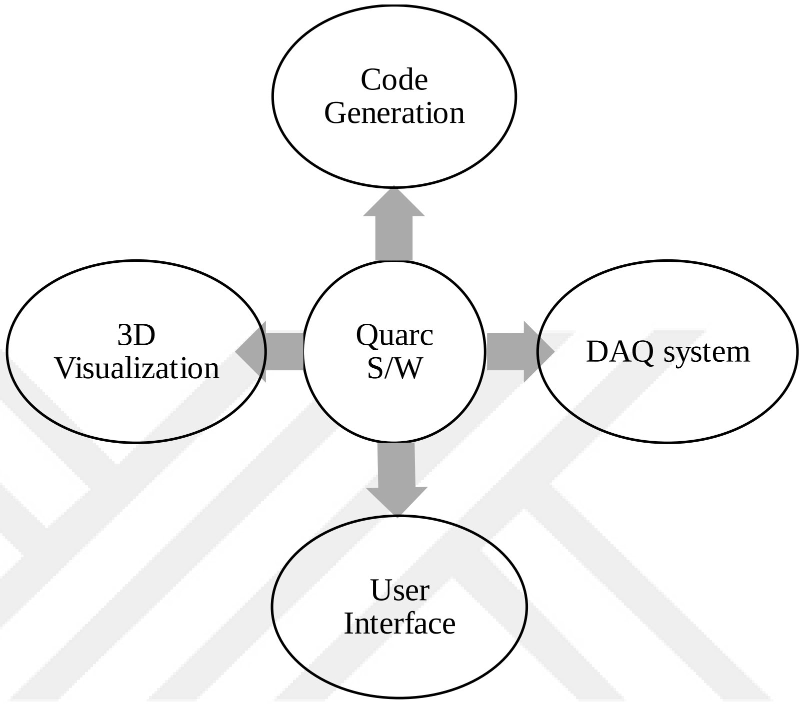


Figure 3.3 Quarc control software structure

Quarc generates real-time code from the standard Simulink blockset. The "*Denso Read block"* is used to read joint encoder positions, effort, status, and error signals for the robot. The "*Denso Write block"* is used to send joint position and joint velocity commands as well as joint PID and feed forward gains for the robot, depending on the block control mode. The robot commands are either desired joint encoder positions or desired change in joint encoder positions (delta positions). The block time cycle of the system is 1ms according to periodic sample times. Figure 5.3 shows the system signals of read and write blocks.

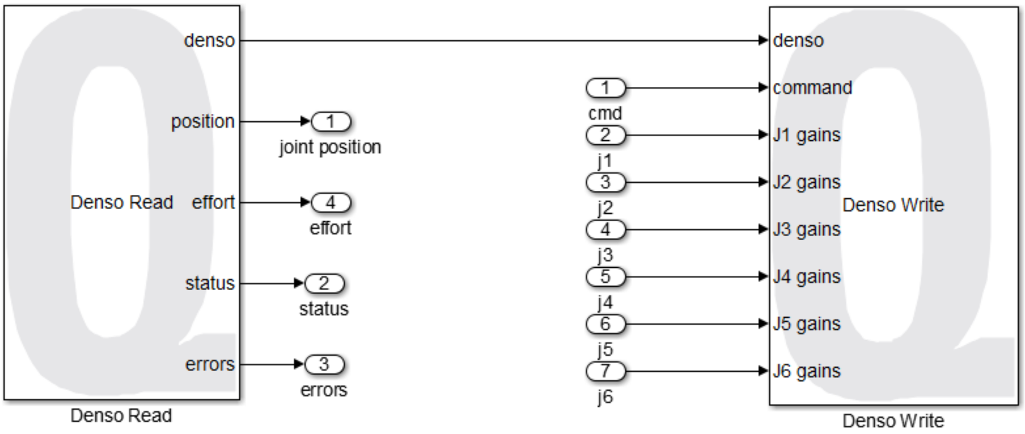


Figure 3.4 System signals of read and write blocks

Referring to Figure 3.4, the command signal is used to apply motion to the system. It contains 6 vectors for the joint position or joint velocity if the velocity mode parameter is activated. The J(1:6) gains are the controller vector gains [P I D F], where P is the proportional gain, I is the integral gain, D is the derivative gain, and F is the feed forward output. The angle of joints are calculated from the joint encoder position unit. Conversion factors are used to convert these encoder positions into joint angles. Effort signal is a feedback vector containing the joint control current (Ampere) without feed-forward current. The "*Error*" signals are 6 elements vector containing the joint axis error codes. The type of error can be addressed in the Status signal which has 6 elements vector containing the joint axis state codes, the Status cases are shown in Table 3.2. The robot communicates with Quarc external Simulation mode through a real-time-capable point-to-point network link. The exchanged data are transmitted via Ethernet protocol.

Simulink library contains the controllers and kinematics for the robot and the tool actuator. It is executed on the slave PC. It communicates with the master robot PC. In addition, the robot can be controlled in both joint level and workspace level as shown in Figure 3.5. In the joint level control, q includes 6 elements vector joint angles input. It contains the joint position in rad, while in the workspace level control. xdot_cmd is 6 elements vector containing robot workspace speed command in m/s. qdot_cmd is 6 elements vector containing the joint speed commands in rad/s.

Table 3.2 Description of Status feedback signal vector

Step No.	Status	Description of System Status
0	Not initiated	Communication problems.
1	Disconnection	Host is not connected.
2	Shutdown	Robot in Error condition.
3	Enable Power	Waiting for system power.
4	Power On	Waiting for system power up.
5	Not Home	Encoders are not initialized.
6	Homing	Perform Homing for system.
7	waiting	Preparing for normal motion.

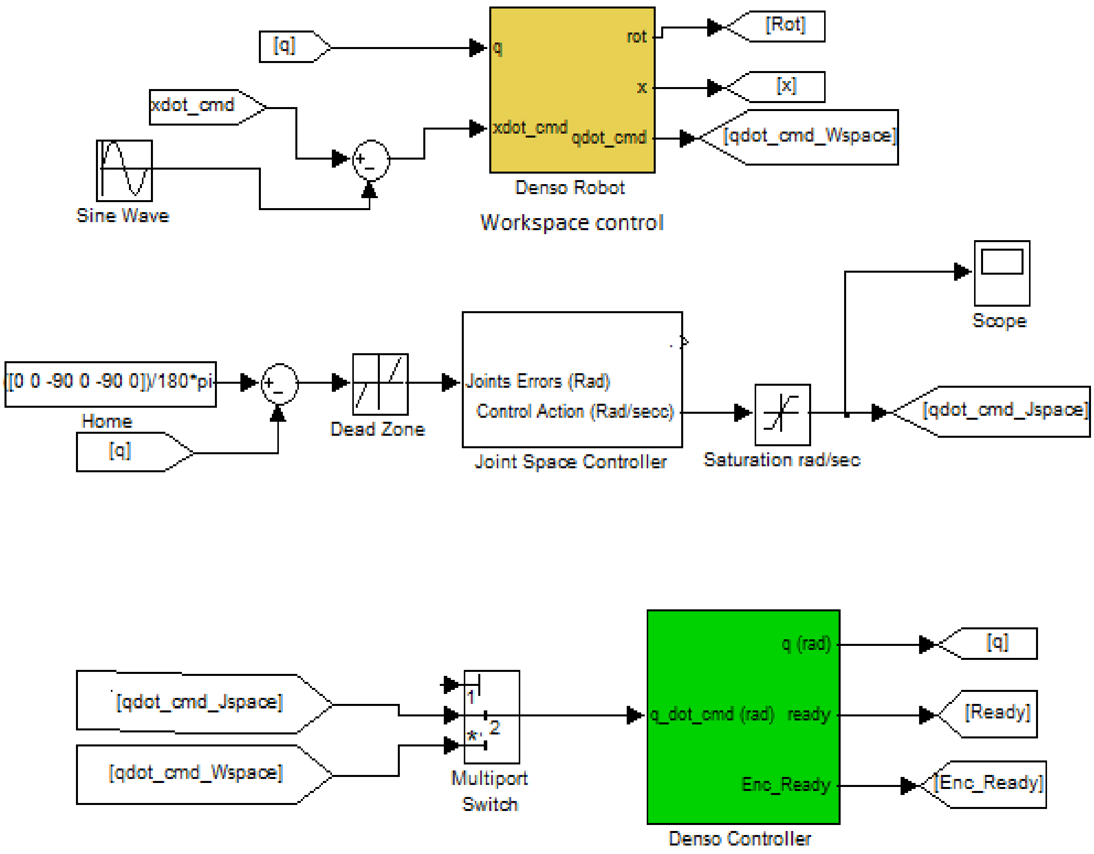


Figure 3.5 Robot control joint level and workspace level

3.4 Force/Torque Sensor

The robot has been equipped with a 6-axis Force/Torque sensor system to measure loads and torques in any direction and orientation over the end effector. It provides robot with the sense of touch and sense of force feel and allow robot to do application of human robot interaction, haptic sensing and controlling for robotic surgery. Force/Torque sensor is a multi-axis load cell transducer and intelligent data acquisition system. Type of this sensor is Gamma FT15681 from ATI Industrial Automation. It measures forces and torques in any orientation over the wrist of robotic arm. The rated forces are Fx, Fy ±130 N and Fz ±400 N. The sensor is communicated with MATLAB/SIMULINK through PCI card to read analog channels which are related to load cell signal [67]. Figure 3.6 shows mounting and coordinate representation of F/T sensor on the system.

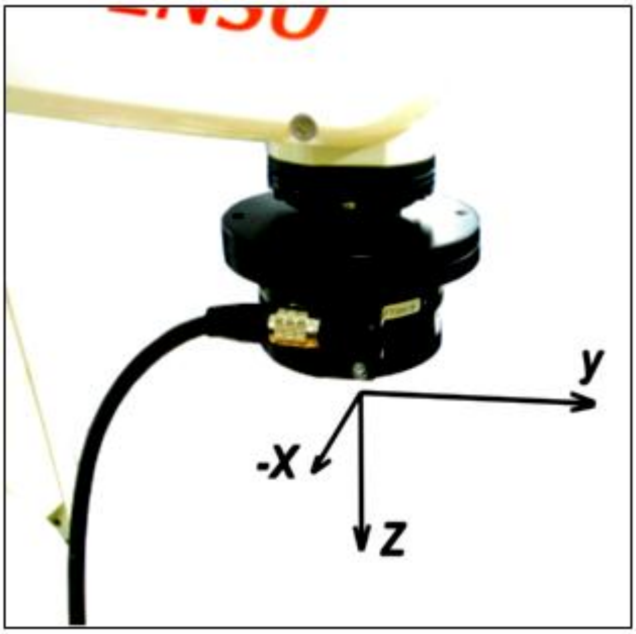


Figure 3.6 Force-torque sensor mounting and sensor coordinates [67]

The force and torque signals are fed to computer program to associate with control part as force sensing. Force/Torque sensor system (F/T) is a multi-axis transducer system that simultaneously converts mechanical loads into forces and torques. A semiconductor strain gauges are attached to inside beams and are considered strain-sensitive resistors. The force is applied to the transducer flexes three symmetrically-placed beams using Hooke’s law:

$$s = E \cdot e \quad (3.1)$$

Where; s is stress applied to the beam (s is proportional to force), E is the Elasticity modulus of the beam, and e is the strain applied to the beam. Figure 3.7 shows the applied force and torque vectors on the transducer [68].

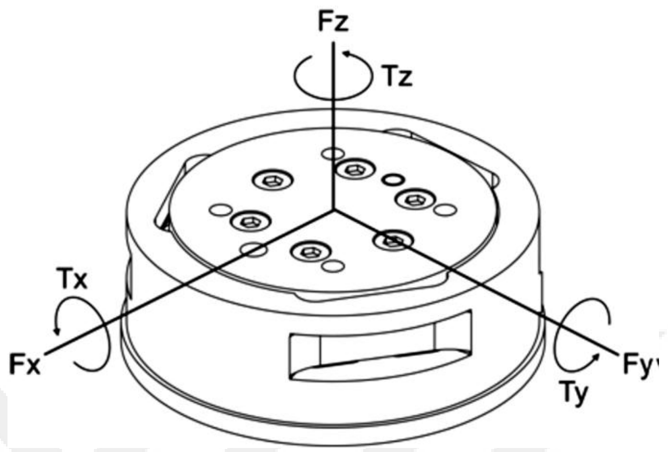


Figure 3.7 Applied force and torque vectors on the transducer [68]

The F/T system communicates via Ethernet IP. The F/T transducer is used as a wrist transducer mounted between a robot and a robot end-effector. Figure 3.8 shows F/T sensor system. The outputs of the sensor are six channels of strain gauge voltages. The outputs are digitized and converted to the force and torque by matrix calculations. The F/T sensor has a compensation matrix. It is used to find the influence between components Fx, Fy, and Fz . The raw signals Uf_x, Uf_y, Uf_z are measured in mV [69, 70].

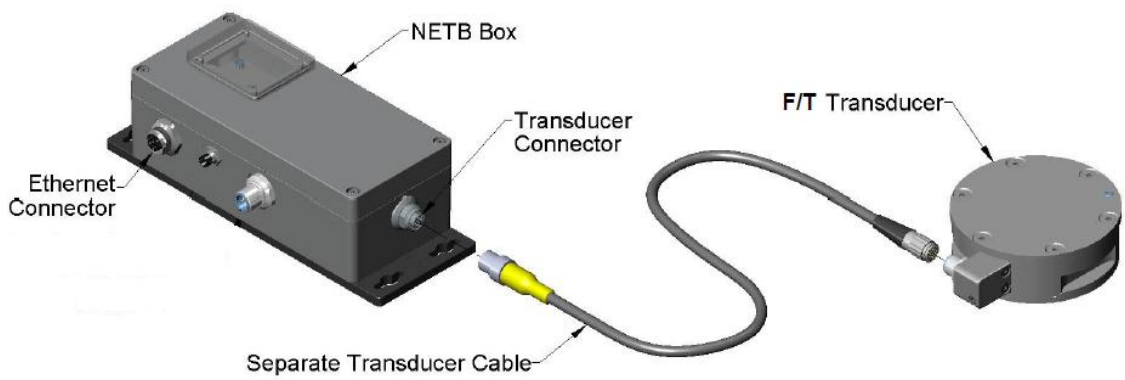


Figure 3.8 Force/Torque (F/T) sensor system [66]

The force and torque values in x, y and z directions are measured from the multiplication of input quantities by the compensation matrix. A compensation matrix coefficients are given as.

$$\begin{pmatrix} Fx \\ Fy \\ Fz \\ Mx \\ My \\ Mz \end{pmatrix} = \begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} & K_{15} & K_{16} \\ K_{21} & K_{22} & K_{23} & K_{24} & K_{25} & K_{26} \\ K_{31} & K_{32} & K_{33} & K_{34} & K_{35} & K_{36} \\ K_{41} & K_{42} & K_{43} & K_{44} & K_{45} & K_{46} \\ K_{51} & K_{52} & K_{53} & K_{54} & K_{55} & K_{56} \\ K_{61} & K_{62} & K_{63} & K_{64} & K_{65} & K_{66} \end{bmatrix} \begin{pmatrix} UF_X \\ UF_Y \\ UF_Z \\ UM_X \\ UM_Y \\ UM_Z \end{pmatrix} \tag{3.2}$$

3.5 Gripper System

The gripper is series of WSG Intelligent Servo-Electric Grippers. It has an electrical system with integrated part detection and force control. External limit switches for detecting the finger position are equipped for automation applications. Figure 3.9 shows the gripper system [71]. Gripper has got 118x32x78 mm dimensions with 550g mass. It can provide 5-50 N grasping force. The gripper offers a number of interfaces, each of which uses the same binary message protocol.

Figure 3.9 Gripper system [71]

Available interfaces are RS-232 serial connection, Ethernet TCP/IP, Ethernet UDP/IP and CAN-Bus. The interface bus is fundamentally different from these interfaces. The interface configuration can be changed using the WSG's web interface. The WSG of the local network or directly computer's network interface and point favorite web browser are connected by IP address, the default http://192.168.1.20 in the address and the command interface is to be chosen by configuring the standard interface. A message command of binary data packet is issued when need to give order for gripper, the gripper message commands are given in Table 3.3.

Table 3.3 Gripper message commands.

Command	State
MOVE	Pre-position movement.
GRASP	Grasp a part.
RELEASE	Release a part.
HOMING	Home the gripper.
STOP	Fast stop.

The gripper communicates with client unit by using binary data packages. All packages start with a preamble signaling the beginning of a new data package. An identification code describes the content of the package. as command ID of the gripper.

Example: an execution command without any return parameters.

>> " Homing " (Command) :-

AAh AAh AAh 20h 02h 00h 00h 00h B3h FDh

Every command is acknowledged by using a standardized acknowledge command according to Table 3.4. The WSG Commander is a software used to get the gripper communication protocol and command set. It allows to send basic commands to the gripper and contains a custom command editor to assemble data packet. Figure 3.10 shows WSG Commander. The data traffic to and from WSG is displayed in communication log as shown in Figure 3.11, so the communication can be understood easily. The software runs on Microsoft Windows operation system.

Table 3.4 Acknowledgment command for gripper.

Byte	Symbol	Description
0..2	PREAMBLE	The begin of a new message and has to be AAAAAAh
3	COMMAND_ID	ID of the command.
4..5	SIZE	Size of the packet's payload in bytes.
6..7	STATUS_CODE	Status code.
8..n	PARAMS	Command specific parameters.

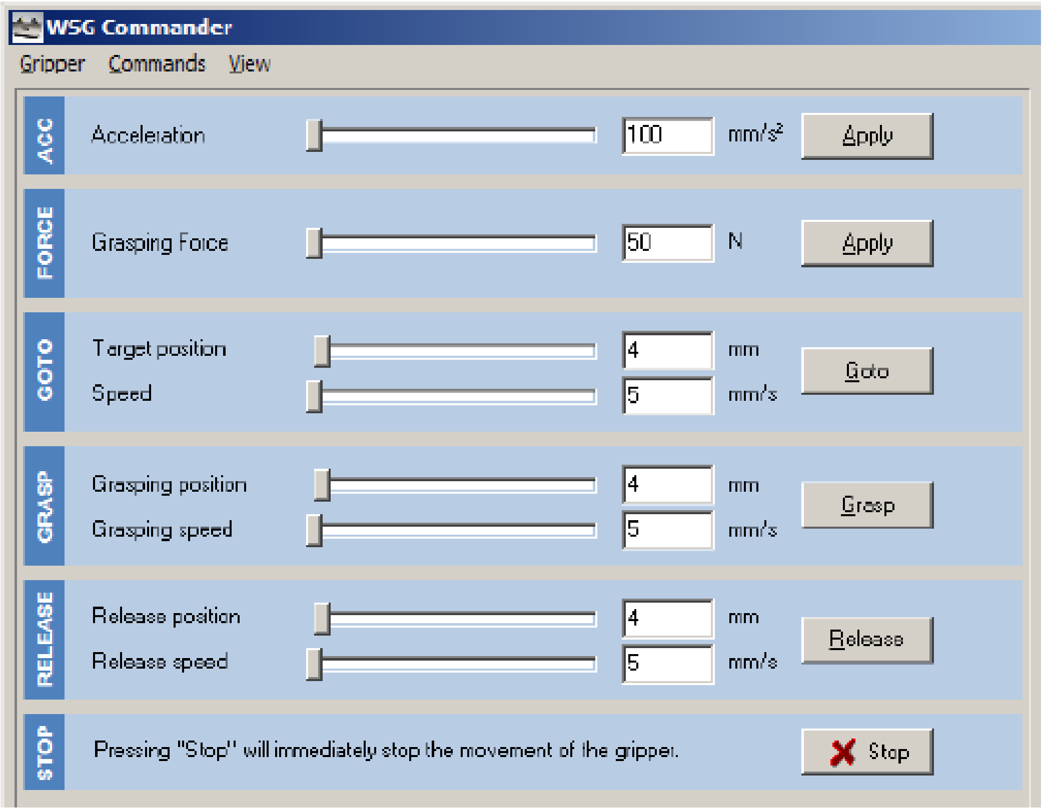


Figure 3.10 The WSG Commander [71]

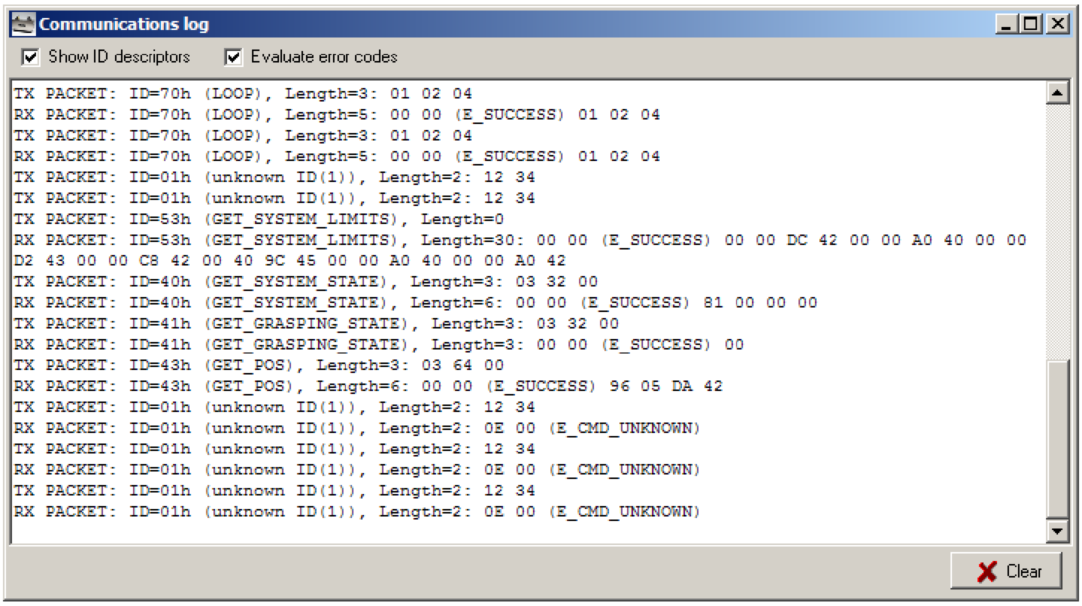


Figure 3.11 Communication Log [71]

3.5.1 Gripper Motion Control

Motion sequence is executed to the reference position of the gripper fingers. The command ‘HOME’ has to be executed prior to any other motion-related command. The direction of homing can be either explicitly specified or obtained from the gripper’s configuration. During homing, the gripper moves its fingers into the specified direction until it reaches its mechanical end stop. The blocking position is used as a new origin for all motion-related commands. Command ID is 20h, command Homing directions are; 0: use default value from system configuration (the default value can be changed via the web interface), 1: Homing in positive movement direction, and 2: Homing in negative movement direction.

i) *Pre-position Fingers* moves the gripper fingers to a defined opening width. This command is intended to pre-position the gripper fingers prior to a grasp. For grasping a part, Grasp Part (25h), absolute movement and relative movement can be selected. The fingers are positioned to the given value in absolute movement. The finger’s opening width is changed relative to their current position in relative movement. The command ID is 21h.

ii) *Grasp* holds a part by passing, it has a nominal width and a speed at which the part should be grasped. When the command is issued, the gripper moves the fingers to the given part width and tries to clamp the expected part with the previously set grasping force. If the gripper can establish the desired grasping force within the defined clamping travel, a part is grasped. If the fingers fall through the clamping travel without establishing the grasping force, no part was found. The clamping travel can be set by using the WSG's web interface. The grasping speed with sensitive parts can be reduced to limit the impact due to the mass of the gripper fingers and the internal mechanics.

iii) *Force* is a force limit used for consecutive pre-positioning and grasping commands. The force limit is the maximum grasping force that is applied on a mechanical contact. Command ID is 32h. Command parameters are; Bytes from 0 to 3 are force limit in Newton, the value is clamped, if it is outside the device's capabilities. The given value is clamped if it is lower than the minimum grasping force and if it exceeds the nominal force.

The robotic system is developed with a new programming method. MATLAB control program is designed to improve the robotic system with the gripper in single control program. MATLAB/Simulink supports communication from the computer to other systems or devices using Ethernet packet. Ethernet is a direct method to send and receive packet with the target application using the Ethernet protocol. A knowledgeable in Ethernet standard is required to provide a dedicated Ethernet connection on computer. The gripper Ethernet card is used for model-based Ethernet communications and computer. Simulink Real-Time model-based Ethernet communication is used to send commands (Home, Move, and Grasp) to the gripper. Figure 3.12 shows designed MATLAB/Simulink Program for Gripper control.

3.6 Complete System Setup

Denso robot, F/T sensor and Gripper are combined together in full system setup to have single MATLAB/ Simulink control program. All devices can be controlled in real time tele-operation. The system has three types of acquisition module, SCHUNK interface module with Ethernet connection for gripper, National Instruments interface module for F/T sensor, and Quanser control module with Ethernet connection for Denso robot. Figure 3.13 demonstrates components involved in a single PC tele-operation setup. Figure 3.14 shows the complete system setup.

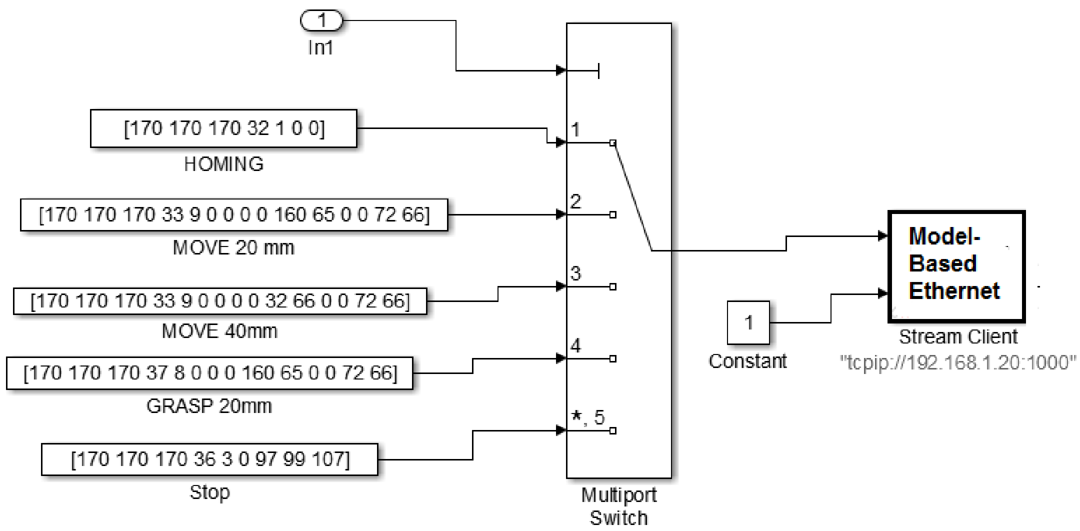


Figure 3.12 Designed MATLAB/Simulink program for gripper control

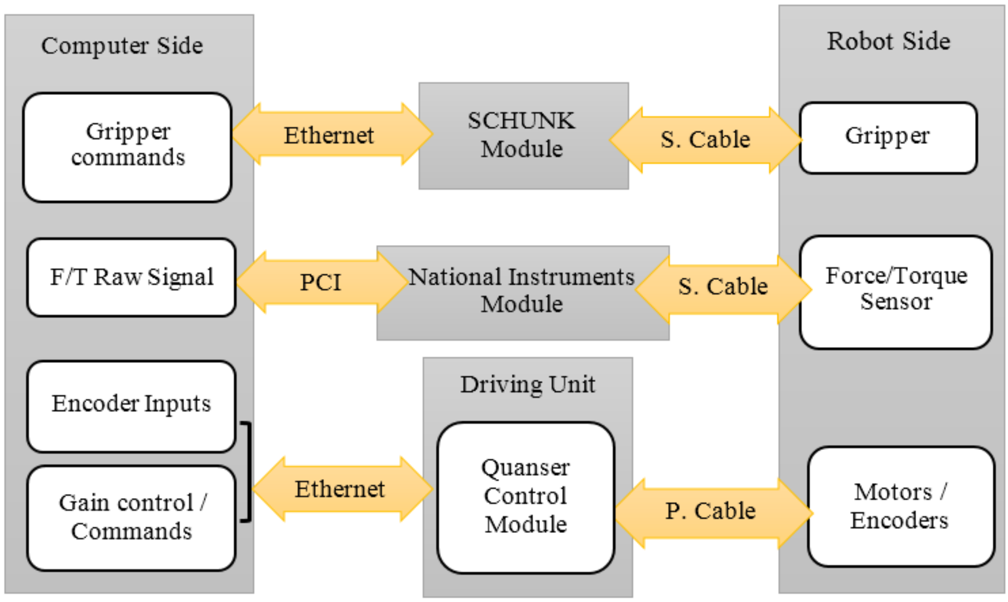


Figure 3.13 Components involved in a single PC tele-operation setup

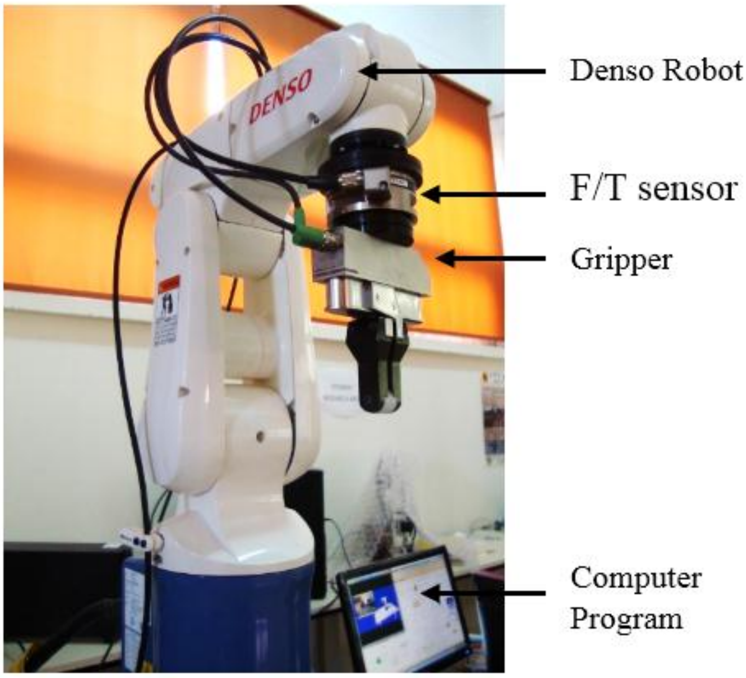


Figure 3.14 Complete system setup

CHAPTER 4

SYSTEM MODELING AND CONTROL METHOD

4.1 Kinematics of Robotic Arm

Fundamentals theory of robotics are used to present the system modeling. The system modeling is the forward kinematics, the inverse kinematics and the dynamical model. The kinematics is based on the homogeneous matrix of current position and orientation of the end effector in an environment. A point in workspace should be able to describe mathematically and by using a coordinate vector. The position of end effector varies with time, the robot should follow different position though desired path, so that a trajectory can generate sequential points with smooth changes from an initial point to a final point.

For six axes robot with revolute joints, it is described by a string “RRRRRR”. A homogenous matrix method of describing the geometry of a serial chain of links and joints is suggested, Homogeneous matrix *A* is 4x4 matrix, which represents rotations *R*, translations *T*, scales *S* and perspectives *P*. In general, the homogeneous matrix represent linear transformation between two positions.

$$A = \begin{bmatrix} R(3,3) & T(3,1) \\ P(1,3) & S(1,1) \end{bmatrix} \tag{4.1}$$

The coordinates of system are suggested for the robot, and supported by recommendations of the Denavit Hartenberg (DH) method [72, 73]. Figure 4.1 shows the DH link parameters.

The following four parameters are known as DH parameters:

d_j : Offset along z_{j-1} from O_{j-1}.

θ_j: Angle about z_{j-1}, between Z_{j-1} and Z_j.

a_j : Distance along X_j from O_j to the intersection of X_j and Z_{j-1}.

αj: Angle about Xj, between Zj-1 and Zj.

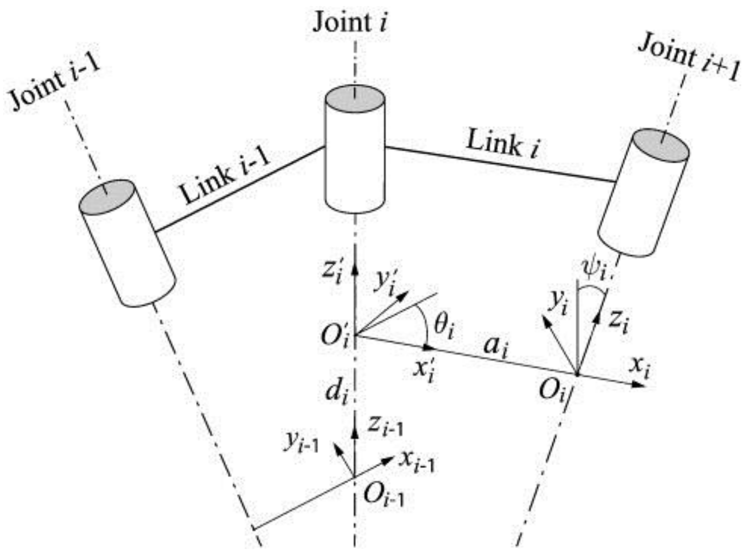


Figure 4.1 The Denavit Hartenberg link parameters [72]

A MATLAB graphical user interface development environment GUIDE program is designed for general six axes robot simulation by using MATLAB Robotics Toolbox. Figure 4.2 shows the designed simulation program by MATLAB/GUIDE to create serial link robot (Link1-6) and control joint angles. The links parameters are entered to program textboxes, then serial link robot is created and plotted by *Plot Robot* command. The slider bars (q1, q2, q3, q4, q5, and q6) are adaptable joint angles to manipulate robot simulation. The motion path can be generated or removed during the process of robot control. The end effector positions (x,y,z) are determined and displayed with respect time. The joints are able to control externally through PC-interface board.

4.1.1 Forward Kinematics and Inverse Kinematics

In forward kinematics; the joint variables of the robot are specified, then the position and orientation of the end-effector are found. If the joint is revolute, the joint variable is the angle between the links. If the joint is prismatic, the extension will be the joint variable. Denavit Hartenberg (DH) is a general method to assign frame parameters of robotic arm. The homogenous transformation matrix T_j refers to the product of four matrixes.

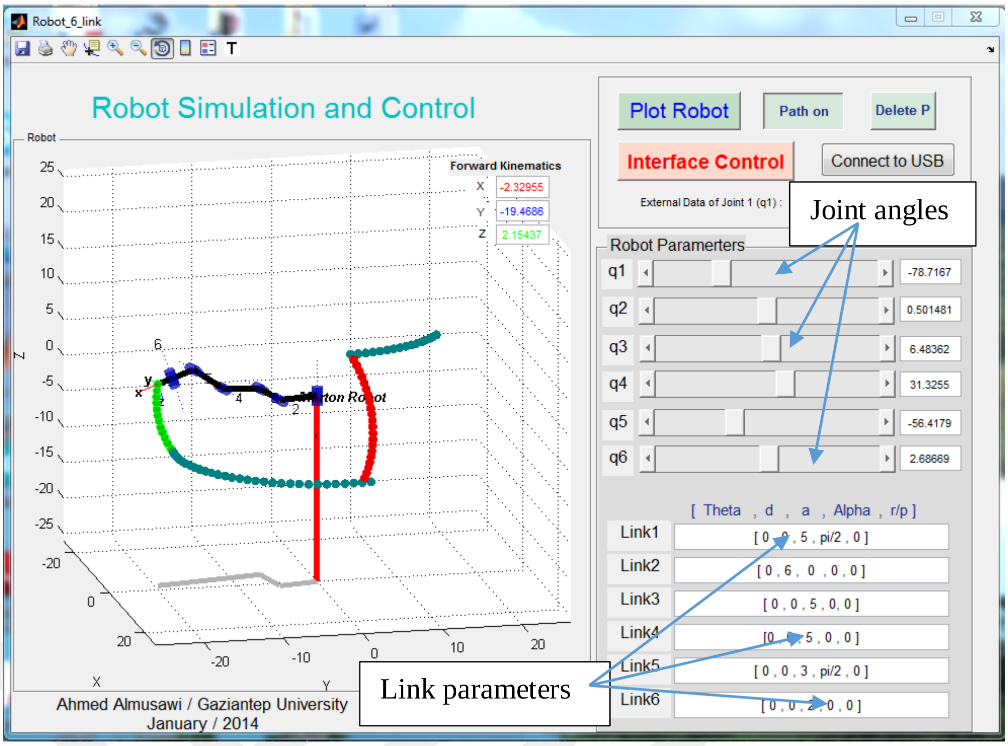


Figure 4.2 Robot simulation and control by MATLAB/GUIDE

$$T_j = Rot_{z,\theta j}\, Trans_{z,dj}\, Trans_{x,aj}\, Trans_{x,aj} Rot_{x,\alpha j} \tag{4.2}$$

$$T_j = \begin{bmatrix} C\theta_j & -S\theta_j & 0 & 0 \\ S\theta_j & C\theta_j & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_j \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_j & -S\alpha_j & 0 \\ 0 & S\alpha_j & C\alpha_j & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

$$T_j = \begin{bmatrix} C\theta_j & -S\theta_j C\alpha_j & S\theta_j S\alpha_j & a_j C\theta_j \\ S\theta_j & C\theta_j C\alpha_j & -C\theta_j S\alpha_j & a_j S\theta_j \\ 0 & S\alpha_j & C\alpha_j & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.4}$$

Where i is the link number, $S\theta_i = sin\theta_i$, $C\theta_i = cos\theta_i$ and θ_i is the joint rotation angle, a_i is the length of links, α_i the twist angles, d_i are the link offsets, and θ are the joint angles.

Inverse Kinematics analysis finds the joint angles for the required position and orientation in Cartesian space. Inverse kinematics solution is more difficult than forward kinematics. There is no global analytical solution method. Each manipulator needs a particular method considering the system structure and its restrictions. Two solution approaches are applied, geometric and algebraic are used for deriving the inverse kinematics solution. The Euler angle parameterization is used to solve the orientation problem. Figure 4.3 shows MATLAB/GUIDE for forward and inverse kinematics of 3 axis robot.

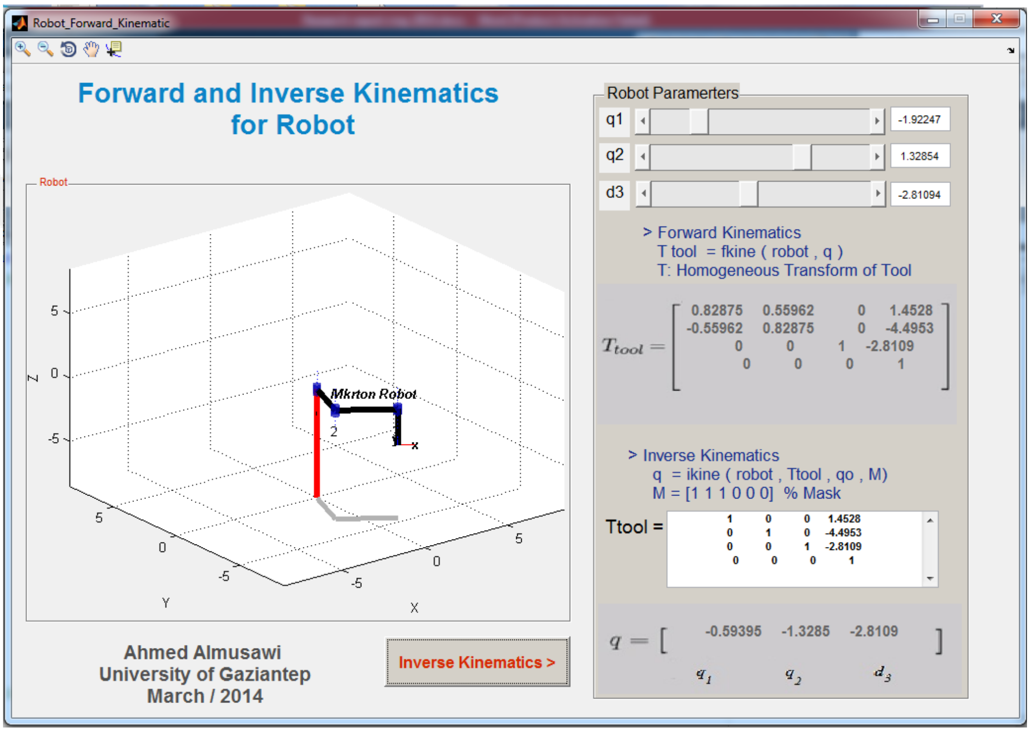


Figure 4.3 Forward and inverse kinematics of 3 axis robot

4.1.2 Trajectories

The movement of end-effector smoothly from initial position to final position is very important in robotics application. A trajectory is a path with specified timing from initial to a final configuration that satisfies specified constraints over points of motion (e.g., velocity and/or acceleration constraints). Figure 4.4 shows an example for application program. The GUIDE program is written to generate a straight lines trajectories. The joint space trajectories can then be calculated. Simulation for robot is done to move robot from initial position to final position in Cartesian space.

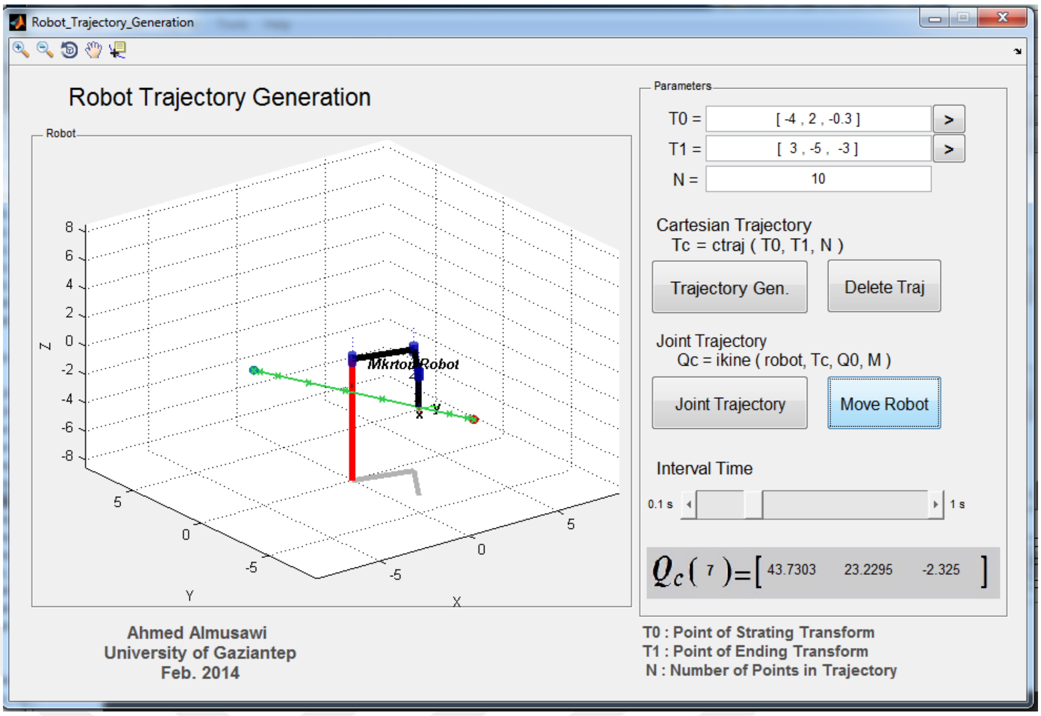


Figure 4.4 Generation of straight line trajectories

Most applications of robotics are to find trajectories through more intermediate points without stopping. A polynomial function of time can be defined to compute and provide the required smoothness with required conditions. A quantic (fifth-order) polynomial is often used:

$$q(t) = At^5 + Bt^4 + Ct^3 + Dt^2 + Et + F \quad (4.5)$$

First and second derivatives give the angular velocity and acceleration.

$$\dot{q}(t) = 5At^4 + 4Bt^4 + 3Ct^2 + 2Dt + E \quad (4.6)$$

$$\ddot{q}(t) = 20At^3 + 12Bt^2 + 6Ct + 2D \quad (4.7)$$

Figure 4.5 shows the robot trajectory in curve motion. The joint angles (Q), velocities (QD) and accelerations (QDD) are plotted through trajectory as shown in Figure 4.6. The initial and final velocity and acceleration are all zero by the default, but the initial and final velocities can be set to non-zero values.

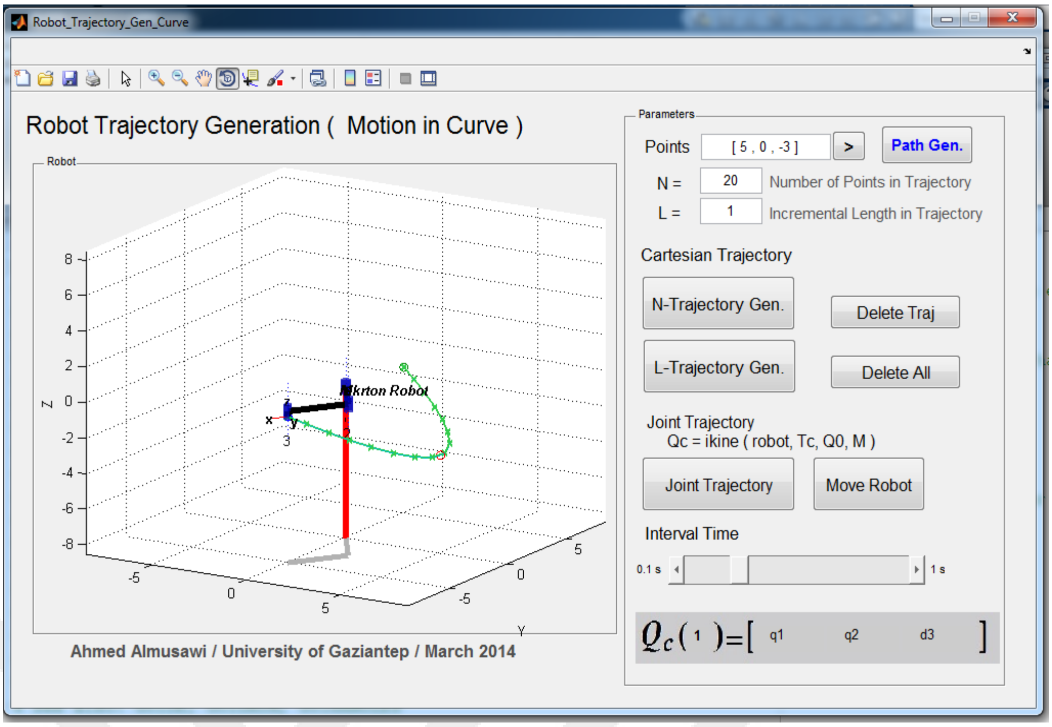


Figure 4.5 Robot trajectories in curve motion

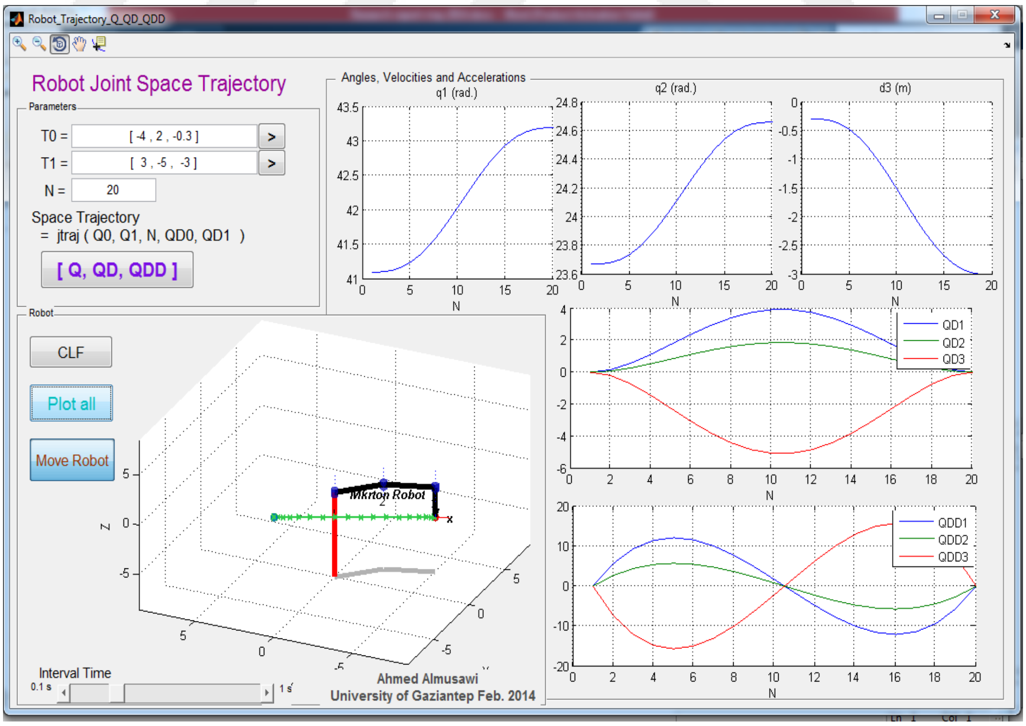


Figure 4.6 Joint angles, velocities, and accelerations through trajectories

4.1.3 Robot Kinematics in Simulink

Robot kinematic analysis is also implemented in MATLAB / Simulink® by using Robotics Toolbox. A block diagram of robotic arm model is shown in Figure 4.7. The parameters of the joint trajectories block contains the initial and final values for the joint coordinates and the duration of the motion. The joint angles are connected to a plot block which will animate a robot in a separate window. The forward kinematics block is used to compute the forward kinematics. Both the plot and forward kinematics blocks have a parameter which is a *SerialLink* robot name. The Cartesian position of the end-effector positions are extracted using the homogenous transformation matrix to x,y,z position (T2xyz block). Figure 4.8 shows the robot presentation in Cartesian workspace. The x-y Graph block plots the projection of motion in x,y coordinates as shown in Figure 4.9. The trajectories of velocity and acceleration are shown in Figure 4.10.

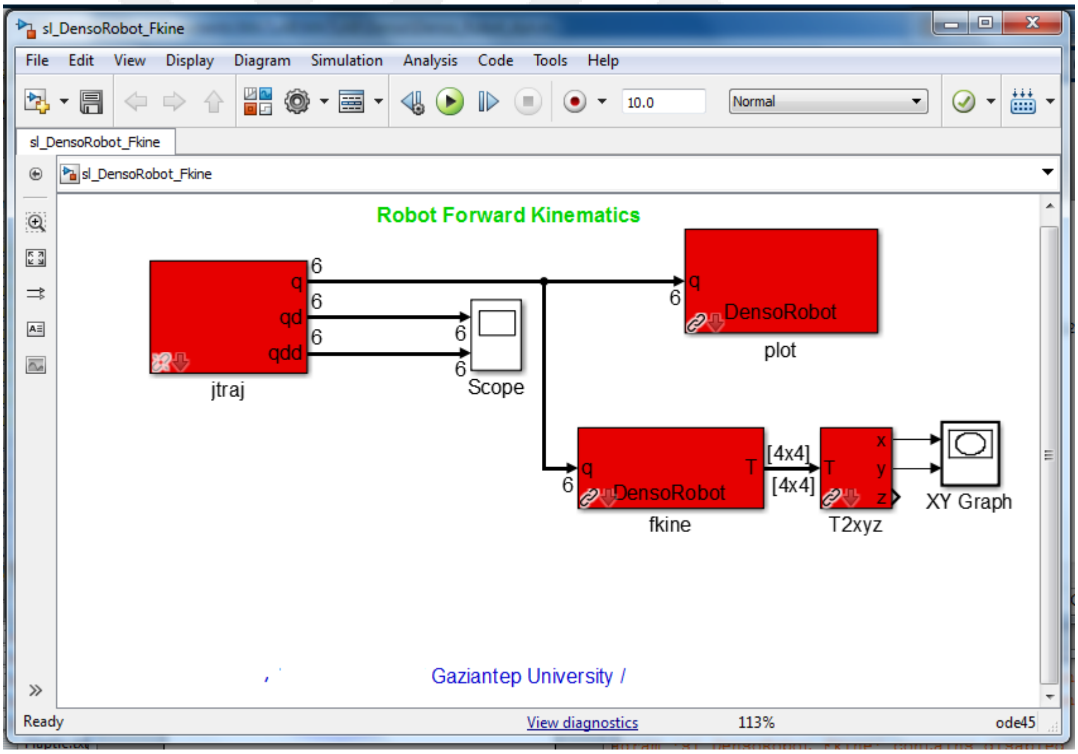


Figure 4.7 Block diagram of robotic arm model

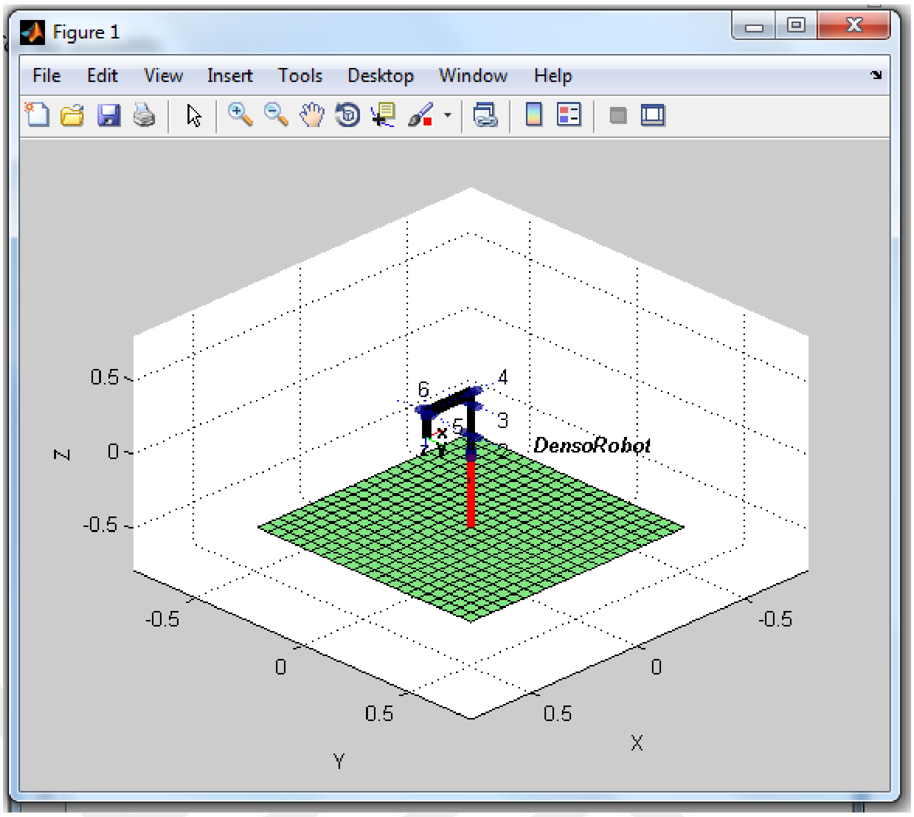


Figure 4.8 Robot representation in Cartesian workspace

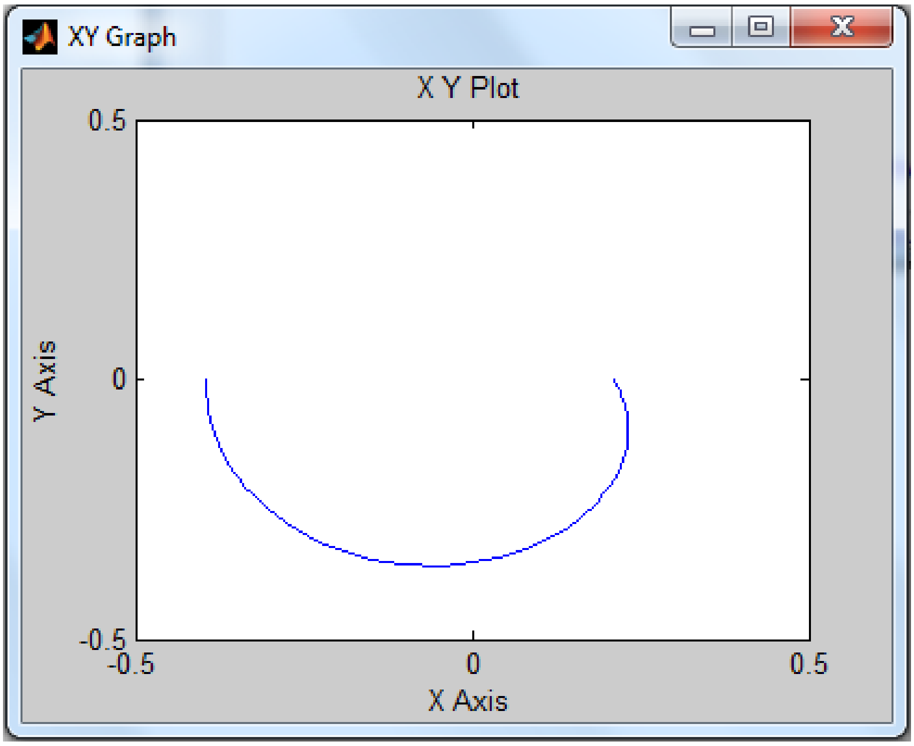


Figure 4.9 Projection of motion in x,y coordinates

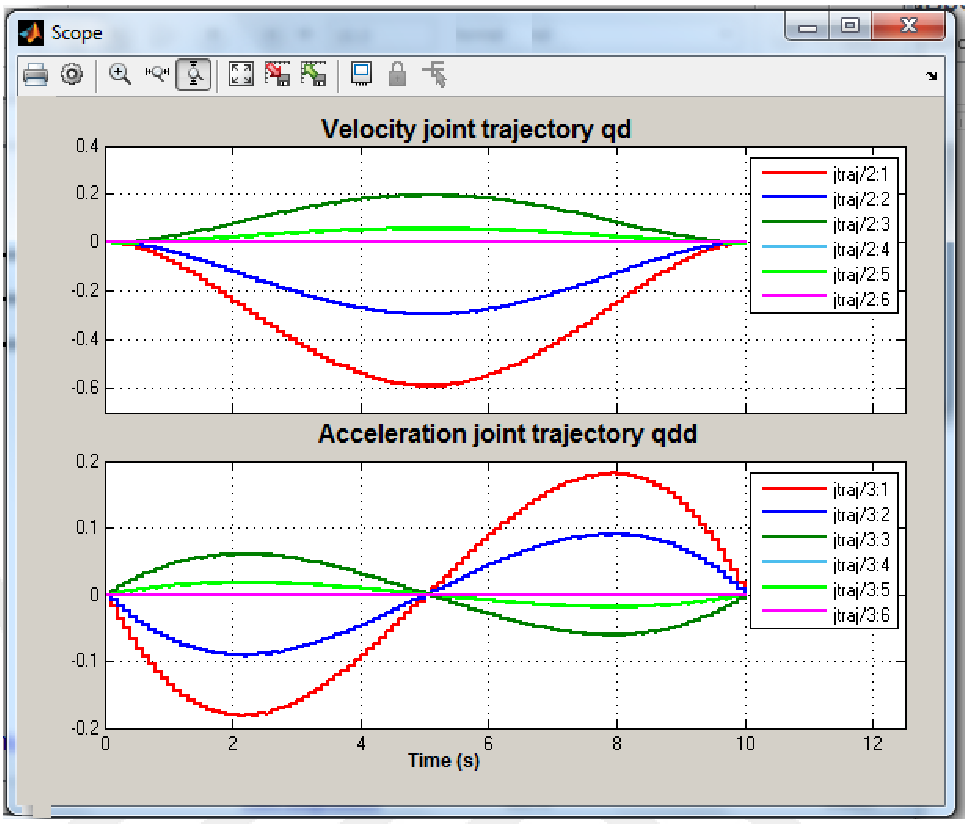


Figure 4.10 Trajectories of velocity and acceleration

4.2 System Model Design

In this section, system model is designed for Denso robotic arm. For the requirement of motion control, the kinematic analysis, and the dynamic model of the system are presented. Virtual reality model is combined with the system dynamic model to build simulation system environment.

4.2.1 Kinematics System Model

The kinematic model of robotic arm gives the relationship between joint variables, position, and orientation of robot's end effector while the dynamic modeling is important for system force/torque control. The coordinate frame is suggested to the system. The coordinate O_0,x_0,y_0,z_0 is fixed to the base which is the base frame and the other coordinate frames are attached to the corresponding links. The reference coordinates frame is shown in Figure 4.11.

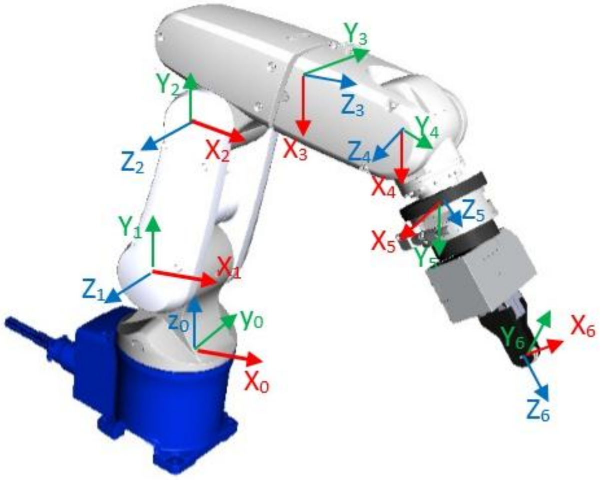


Figure 4.11 Reference coordinates of the system

The homogeneous transformation matrix is stated to represent the position and orientation of end effector with respect to base coordinate. A homogeneous transformation matrix T_6^0 for overall system is given:

$$T_6^0 = \begin{bmatrix} R_6^0 & P_6^0 \\ 0 & 1 \end{bmatrix} \quad (4.8)$$

Where R_6^0 is a 3x3 rotation matrix, and P_6^0 is a position vector of the end-effector in the base frame coordinate. The Denavit Hartenberg (DH) method is applied to the Denso robot. The robot's transformation matrices have been denoted. The DH parameters of the robot are shown in Table 4.1.

Table 4. 1 DH parameters of the system

Link i	$\boldsymbol{\theta_i}$	$\boldsymbol{d_i}$	$\boldsymbol{a_i}$	$\boldsymbol{\alpha_i}$
1	q1	0.125	0	Pi/2
2	q1	0	0.21	0
3	q3	0	-0.075	-pi/2
4	q4	0.21	0	Pi/2
5	q5	0	0	-pi/2
6	q6	0.07	0	0

The system has six links and a gripper, so that the homogeneous transformation matrix is calculated by multiplication of matrices as follows:

$$T^0_{gripper} = A_1\, A_2\, A_3\, A_4\, A_5\, A_6\, A_{gripper} \tag{4.9}$$

Where

$$A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & .125 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 0 & 0 & .21 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_3 = \begin{bmatrix} 1 & 0 & 0 & -.075 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & .21 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & .17 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T^0_{gripper} = \begin{bmatrix} -1 & 0 & 0 & .21 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & .24 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.10}$$

Where $A_{gripper}$ is the transformation matrix of the gripper, for any identified joint angles and gripper transformation matrix the forward kinematics can be calculated. The inverse kinematics has got several solutions for each of the joint angles. These are corresponded to many different robot configurations such as “elbow up”, “elbow down”, “wrist up”, “wrist down”, “shoulder forward”, and “shoulder back”. The program chooses the closest solution to the previous joint angles.

4.2.2 Dynamic System Model

The dynamic system is composed of a six individual links. The dynamic equation of the robotic system can be written as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + J(q)^T g = U \tag{4.11}$$

Where the $M(q)$ is the robot inertia matrix, $C(q,\dot{q})$ is the Coriolis and centripetal coupling matrix, $F(\dot{q})$ is the friction force, $G(q)$ is the gravity loading, $J(q)$ is the manipulator Jacobian, g is the joint forces due to a wrench applied at the end effector,

U is the vector of torque related with the joint coordinates q, and $\dot{q}, \ddot{q}$ are joint velocity and acceleration [74]. The system model of the robotic system is designed by SimMechanics / MATLAB, the mechanical parameters of links are taken from Denso robot specification. This model is a nominal system that assists controller to find effective force and torque. Figure 4.12 presents the dynamic model of six individual links. The sub system model has a single link model. The single link components are the revolute joints, bodies, sensors and actuator as shown in Figure 4.13. The system receives the torques as input while the outputs are the joint angles, angular velocities, and joint torques. A six axis F/T sensor is also added to the system model to simulate the work of F/T sensor in real system. The system model is operated in the same time with the real robotic system. The sensor in this model gives the nominal conditions for forces and torques F_g, T_g which are caused by the inertia and weight of the gripper during motion.

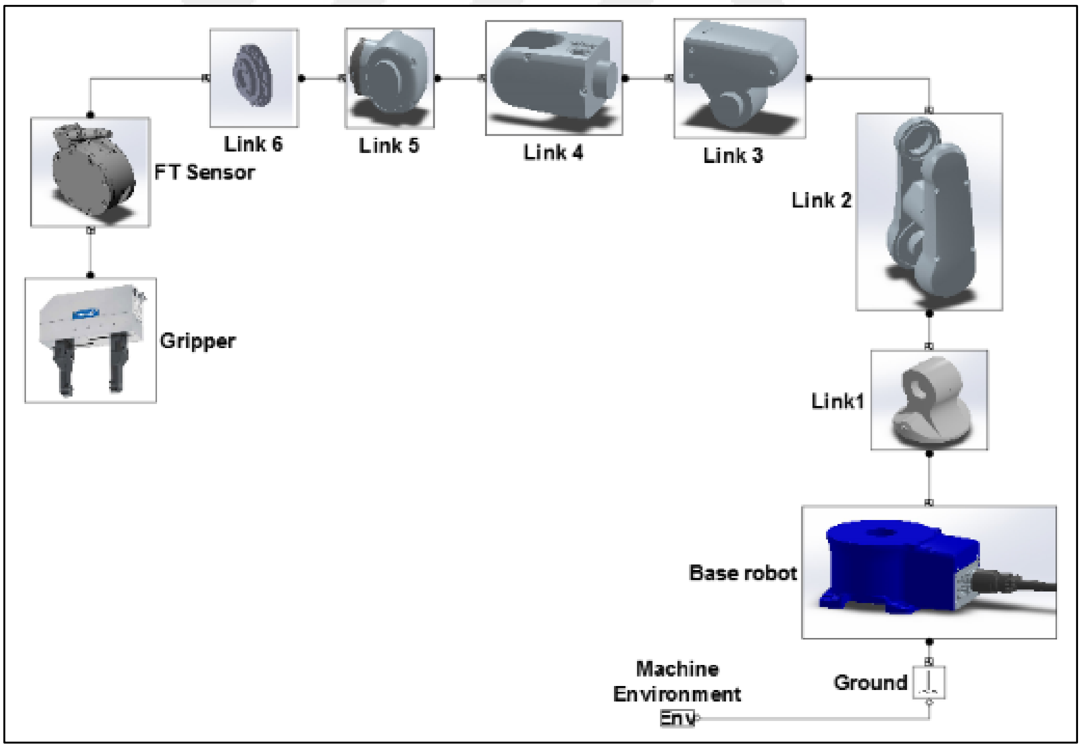


Figure 4.12 Individual links of system of system dynamic model

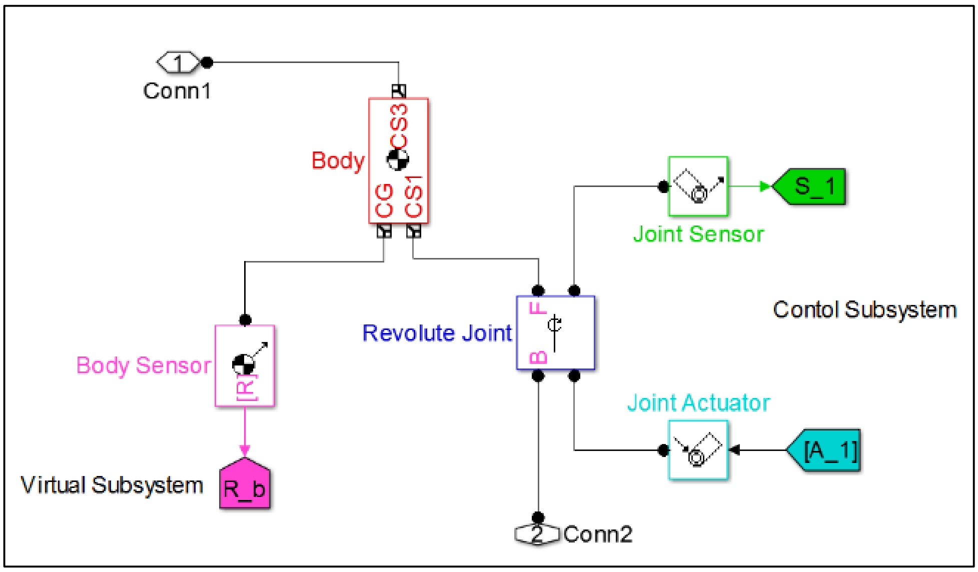


Figure 4.13 Single link components of subsystem model

The actuators of robots joints are applied as a transfer function which has been estimated from the input / output (I/O) system response by using System Identification Toolbox. The actuators of robot’s joints are applied as a function which is identified from the system response characteristics as shown in Figure 4.14.

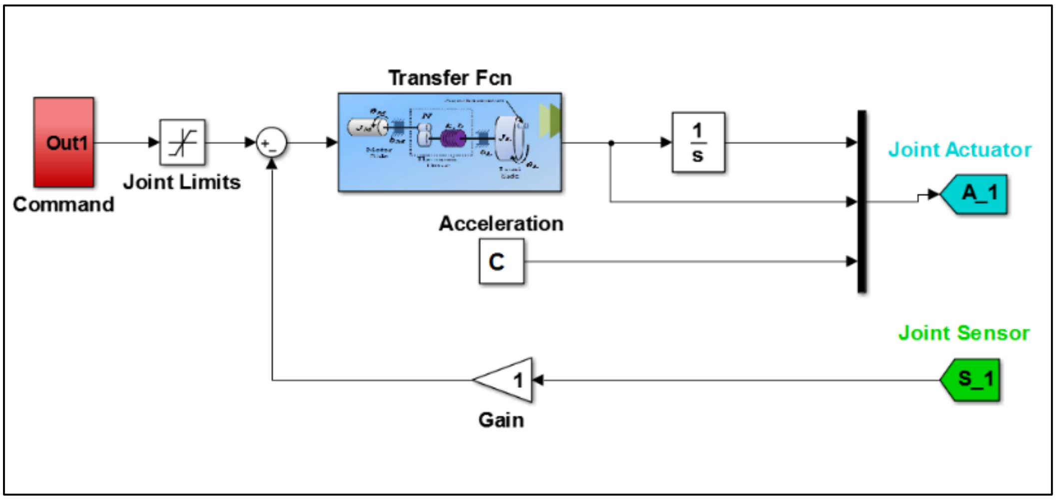


Figure 4.14 Modeling of actuator for the system

4.2.3 Virtual Reality Model (VRM)

The virtual model is assembled from SolidWorks drawings for the robotic system. F/T sensor, and gripper are included. The virtual model is connected with Simulink to visualize the system by using a Simulink/ 3D animation. The dynamic system behavior is verified in a virtual reality environment. The link's shape, geometry, position and rotation are designed by the Virtual Reality Modeling Language (VRML) to have real time animation for the system. The virtual model is combined with the system model to create a simulation environment for the system. The simulation system demonstrates the behavior of system during experiments. Figure 4.15 shows the virtual model of robotic system.

Figure 4.15 Virtual reality model of the robotic system

4.3 Proposed Control Method

The physical interactions of human hand and robot produce a haptic force and torque on the end effector. They are measured by F/T sensor while the end effector itself exerts inertial force and torque on the F/T sensor. Referring to Figure 4.16, F/T sensor measures the gravitational force/torque of the end effector (gripper) and human hand external force/torque. The resultant signals measured by F/T sensor are as follows:

$$\begin{bmatrix} F^s \\ T^s \end{bmatrix} = \begin{bmatrix} F_{app} \\ T_{app} \end{bmatrix} + \begin{bmatrix} F_g \\ T_g \end{bmatrix} \tag{4.12}$$

Where F_{app}, T_{app} are the vectors of the applied force/torque acting on the gripper. F_g, T_g are the vectors of force and torque from the mass of gripper, which is a simple force of the gravity load when the robot is moving slowly, but it becomes more complex during motion. The gripper inertia effect and moment of inertia effect are added to increase the measured force and torque. F^s , T^s are the vectors of the resultant force and torque measured by F/T sensor. The inertia of the gripper gives undesirable effect which is unavoidable since all materials have a mass, the inertia effect causes position error during task execution [30].

The second problem is the sensor's axes rotation during motion, so that the force components at the coordinate axes of the sensor vary during robotic manipulation.

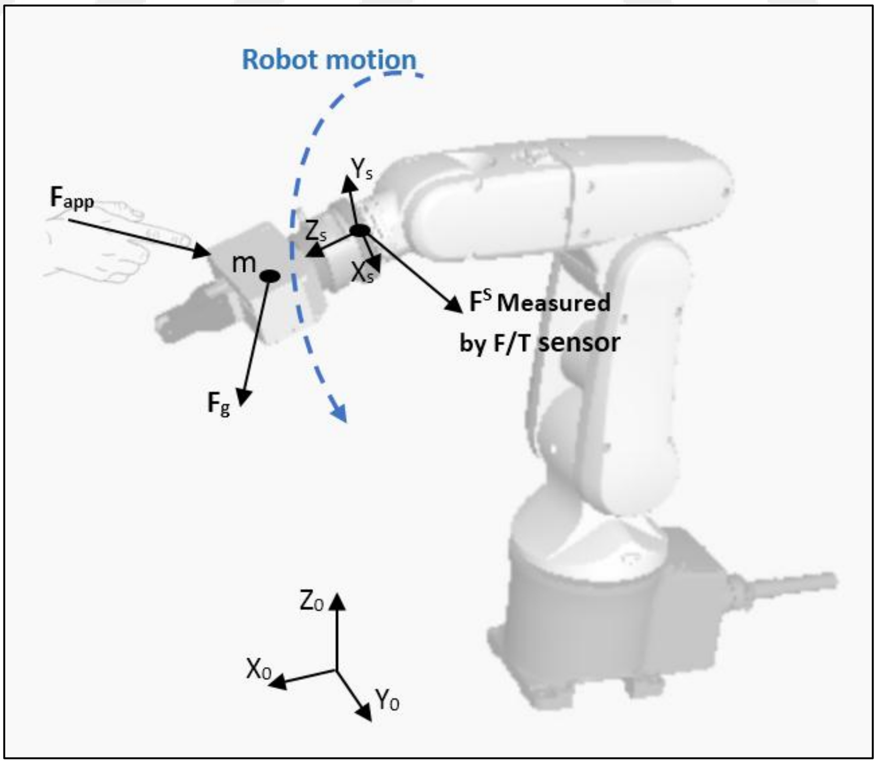


Figure 4.16 Forces acting in the system

4.3.1 Force to Position Control Method (F/P)

The virtual spring control method was used in many studies to obtain the relation between force and position [29, 75, 76]. In this study the method is modified to include the tool inertia effect compensation and torque to orientation. The proposed method is used to obtain the effective value of human haptic force and torque and convert them to the position and orientation which are required during teaching by HRI. The movement response in robot is proportional to the amount of the applied force, the force is in a three dimensional axes and it is relative to the coordinate of F/T sensor on the wrist of robot. As the sensor forces vary during manipulations, these forces are then transformed to the robot base coordinate by multiplying with instantaneous inverse rotation matrix of robot as shown in equation (4.13).

$$F^0 = (R_s^0)^{-1}\left[F^s - F_g\right] \tag{4.13}$$

Where F^0 is the force vector relative to base coordinate and R_s^0 is the rotation matrix between F/T sensor coordinate and base coordinate. F_g is subtracted from F^s to obtain human applied force F_{app}. The force F_g from gripper inertia is eliminated. A virtual differential amount in displacement of end effector is calculated as follows:

$$F^0 = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \tag{4.14}$$

$$F^0 = K_{vs}\,\delta P \tag{4.15}$$

K_{vs} is the stiffness of the virtual spring, and δP is the displacement produced by the applied force along the same direction. The input variable is the force and the dependent variable is δP, so equation (4.16) can be written as:

$$\delta P = C_{vs}\,F^0 \tag{4.16}$$

C_{vs} is the compliance of virtual spring and $C_{vs} = 1/K_{vs}$,

$$\delta P = \begin{bmatrix} \delta X \\ \delta Y \\ \delta Z \end{bmatrix} = \begin{bmatrix} F_x C_{vs\_x} \\ F_y C_{vs\_y} \\ F_z C_{vs\_z} \end{bmatrix} \tag{4.17}$$

The C_{vs} values are chosen according to the required response for system and range sensing for F/T sensor. The displacement is added to the previous position of the end effector.

$$P_n = \delta P + P_c \tag{4.18}$$

Where P_n is the next new position and P_c is previous position of end effector which is obtained from forward kinematics for current joint angles $q_{Feedback}$ as shown in Figure 4.17. The inverse kinematics is applied for new position P_n and the new joint angles are sent to robot driving unit, then the joint motor torques T_m are fed to robot and the joint encoder angles *Enc.* are returned back to PC.

4.3.2 Torque to Orientation Control Method (T/O)

The applied torque on robot is converted to the desired orientation of end effector, according to virtual spring relation. A virtual torsion spring is used to represent the relation between the robot rotation angles (roll, pitch, and yaw) and the applied torque signals. The response in rotation angles is proportional to the amount of the applied torque. The applied torque is a three dimension axes, and it's relative to the coordinate of F/T sensor on the wrist of robot. These torques are transformed to the robot base coordinate by multiplying it by inverse rotation matrix of robot as shown in the equations below:

$$T^0 = (R_s^0)^{-1}\left[T^s - T_g\right] \tag{4.19}$$

$$T^0 = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{4.20}$$

Where T^s is the vector of torque relative to sensor coordinate, and T^0 is the vector torque relative to base coordinate. T_g is subtracted from T^s to obtain human applied torque T_{app}, and the torque T_g from moment of inertia of gripper is eliminated. A differential amount in rotation angles, $\delta\theta$ of end effector is calculated as the following:

$$T = K_{tvs}\,\delta\theta \tag{4.21}$$

$$\delta\theta = T/K_{tvs} \tag{4.22}$$

$$\delta\theta = C_{tvs}T \tag{4.23}$$

$$\delta\theta = \begin{bmatrix} \delta Roll \\ \delta Pitch \\ \delta Yaw \end{bmatrix} = \begin{bmatrix} C_{tvs\_x}T_x \\ C_{tvs\_y}T_y \\ C_{tvs\_z}T_z \end{bmatrix} \tag{4.24}$$

Where K_{tvs} is the virtual torsion spring stiffness, T is the applied torque, and C_{tvs} is the virtual torsion spring compliance. $\delta\theta$ is the change of orientation of the end effector that is produced by applying torque on the end effector. The change in orientation is added to the previous orientation θ_c of the robot, so a new end effector orientation angle θ_n is generated.

$$\theta_n = \delta\theta + \theta_c \tag{4.25}$$

The applied force / torque from the user on end effector is a pulse shape signal. Which is a vector quantity. It is increased when the user is applying a touch and decreased when user hand is removed. The force and torque signals are integrated to convert them to scalar quantities.

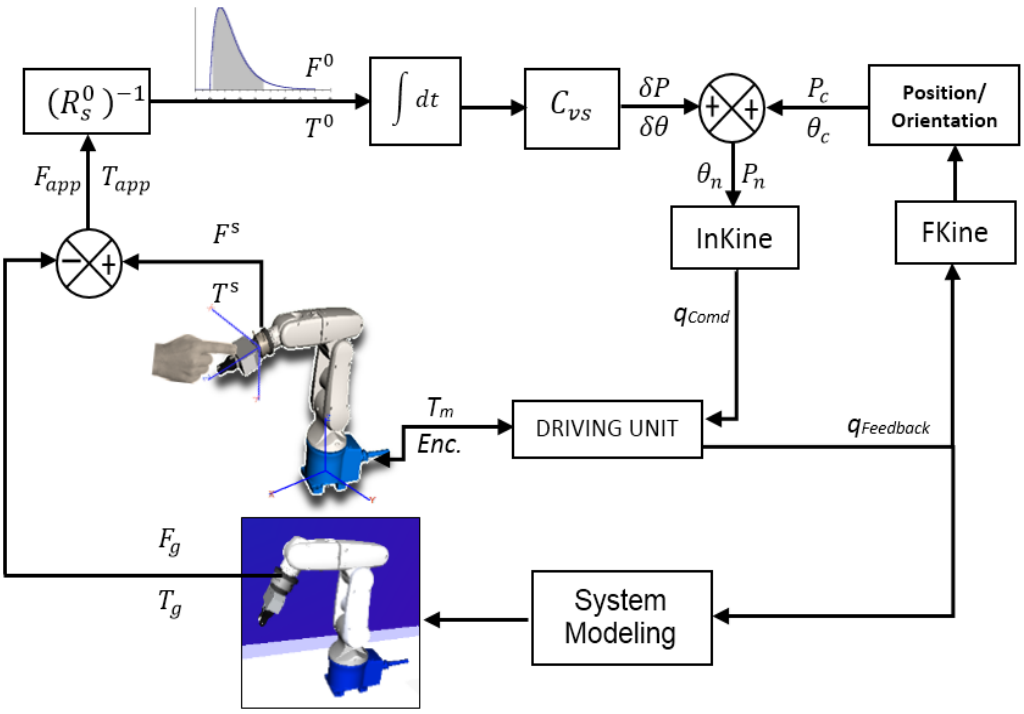


Figure 4.17 Proposed control method diagram

4.4 Artificial Neural Network for Inverse Kinematics Solution

In general, the desired motion of robot is carried out in the Cartesian coordinate, while the robotic arm motion is controlled by joint coordinate. A solution for the inverse kinematics is very important to be calculated. Solving inverse kinematics problem for robotic manipulators is a difficult and also quite challenging task. The difficulty of this problem is given by the robot's geometry and the nonlinear trigonometric equations that describe the relationship between the Cartesian space and the joint space. Although a closed form solution to this problem is preferable in robotic system, sometimes it is impossible to find. Therefore, different methods are used to determine the solution for inverse kinematics problem are studied as geometrical solutions, and numerical algorithms. This task totally depends on the designed structure of the robot while many robots such as redundant manipulators do not have an analytical solution for the inverse kinematics.

In this study, two ANNs are designed for inverse kinematics of robotic arm. The first one is the traditional ANN as used in serial robotics inverse kinematics analysis. The second is the proposed ANN by considering the feedback of current robot configuration (current joint angles) in the design of ANN.

4.4.1 Traditional Design of Artificial Neural Network

A traditional design for ANN is used in many studies [34, 36-38]. In order to utilize the advantages of this proposed method, traditional ANN is designed in this study to solve the inverse kinematics. In this ANN, the elements in the input layer are six variables; which are the position P and orientation R of gripper in Cartesian coordinates. The number of hidden layers is 10. The output layer has six elements of the angles of joint Q. Training, validation and testing are done by using MATLAB/ Neural Network Toolbox. Figure 4.18 shows block diagram for traditional ANN. Its model is given.

$$[Q] = ANN\_Traditional\_Net(P, R) \tag{4.26}$$

The inputs are uniformly enclosed with the workspace of specified position. The corresponding inputs/outputs are computed by solution of forward kinematics. In this way, each position of the robot has a unique joint configuration in the neural network inputs/outputs set. The training algorithm is the Levenberg-Marquardt (LM) back propagation, it is used to assure fast convergence of the training error, and is also a very popular curve-fitting algorithm.

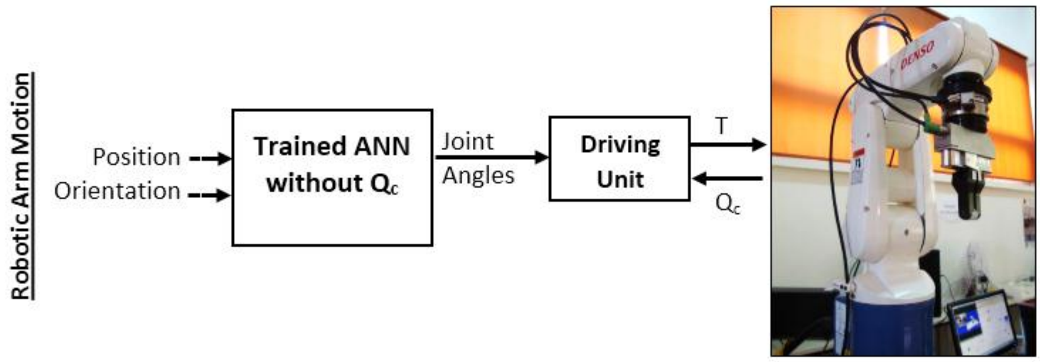


Figure 4.18 Block diagram of robot motion control by traditional ANN

4.4.2 Proposed Artificial Neural Network Design

Robot always starts motion from current position, and in most applications robot moves on its trajectory by using sequential point's path. So the inclusion of current joint configuration in ANN has a positive effect in the estimation of joint angles for the next desired position. Here, a novel neural network design is proposed and used to solve the inverse kinematics problem of robotic arm. The proposed method relies on the constraints of the kinematics of robotic arm to achieve robot's motion in an intelligent way with high accuracy in position. The contribution of proposed design considers the current joint angles of the robot for solving the inverse kinematics of robot by ANN. The inclusion of the current joint angles during training produces a strong network and adjusts the weights with very low error.

Forward kinematics is used to generate the input/target data set which are used in trainings, the inputs of neural network are the desired position/orientation and current robot joint configuration while the targets are the required joint angles of the robot relative to those points. The inputs and outputs of neural network as follows:

$$[Q] = ANN\_Proposed\_Net(P, R, Q_c) \quad (4.27)$$

The proposed neural network has 13 elements in the input layer, which are the gripper position P and orientation R in Cartesian coordinates, and current joint angles Q_c.

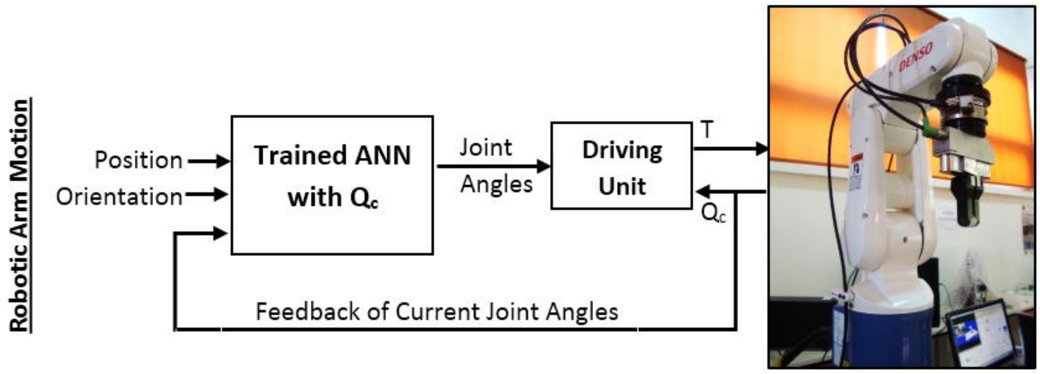


Figure 4.19 Block diagram of robot motion control by proposed ANN

The output layer has six elements of joint coordinates, which are the joint angles *Q* of the robot. Figure 4.19 shows a block diagram for proposed ANN. The performance of the neural network was determined based on the mean squared error MSE between the neural network's actual output and the desired output.

4.5 Robotic Surgery

Robotic medical assistance is the most modern application in minimally invasive surgical techniques; a small incision is made by the surgeon with a precision tool. The robotic systems introduce a surgical control and visualization for the operating field.

4.5.1 Video-Assisted Thoracoscopic Surgery (VATS)

Video-assisted Thoracoscopic surgery (VATS) is a type of thoracic surgery performed using a small video camera that is introduced into the patient's chest via a scope. A surgeon uses Thoracoscope to visually examine pleura, lungs, and mediastinum and to obtain tissue for testing purposes. VATS is tackled in this study as a Robotic Thoracoscopic Surgical (RTS) application. A small incision is opened for Thoracoscopic camera; this technique is called Robotic Minimally Invasive Surgery (RMIS). A special instrument called Thoracoscope is used. This is a thin, tube-like instrument which has a camera built into the end. Thoracoscope is a small surgical scope which magnifies the body's internal structures and projects it onto a video monitor during the operating room [77-79]. The instrumentation for VATS includes the use of a camera-linked 5 mm or 10 mm fiber-optic scope. It can visualize 30º of angle and either conventional thoracic instruments or Thoracoscope. Figure 4.20 Shows Thoracoscopic surgery control points with the instrument.

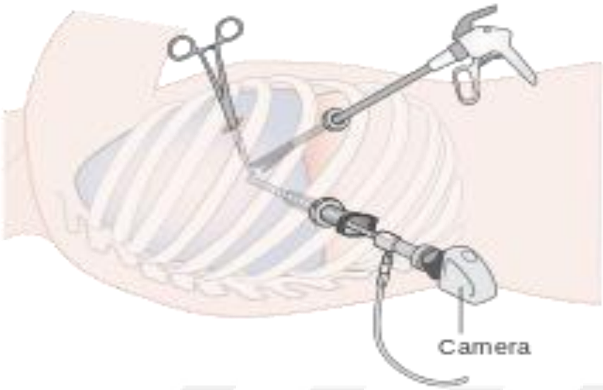
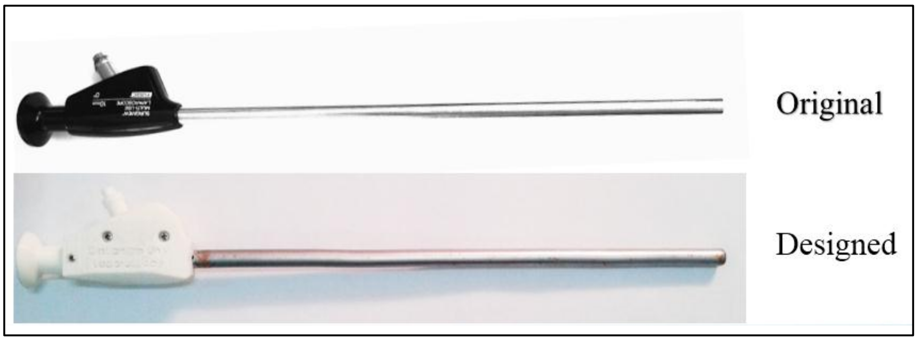
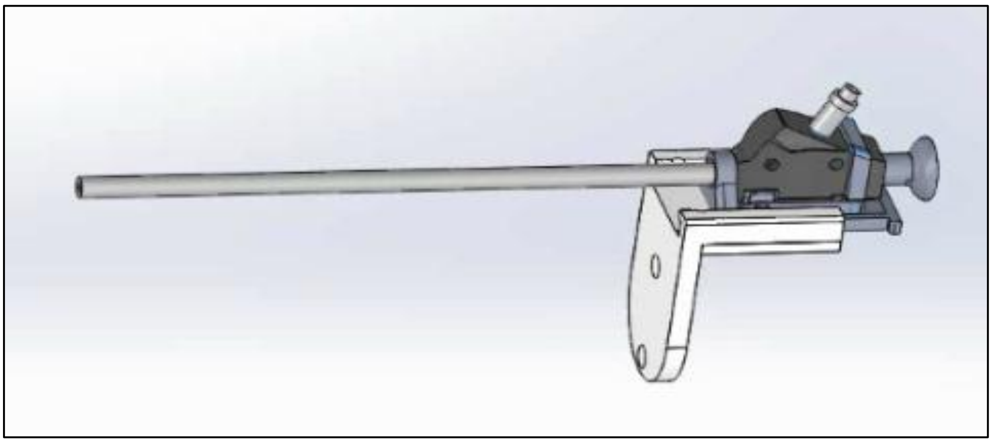


Figure 4.20 Surgery of Thoracoscope [80]

A case study combines Thoracoscopic camera with the robotic arm (Denso Robot) and to investigate how it can be manipulated from the computer. The limitations of Thoracoscopic surgery can be reduced. A holder piece Thoracoscopic device is designed by SolidWorks software and produced by 3d printer. It can be fixed on robot easily and it has capability of being assembled and disassembled by hand in the case of cleaning, replacing, or any other emergency case. Figure 4.21 shows the designed holder with Thoracoscope. Figure 4.22 shows the robotic arm with holder, Thoracoscope, and plastic human body model.



(a)



(b)

Figure 4.21 Thoracoscope, (a) Original and designed Thoracoscope, (b) Designed Thoracoscope with holder.

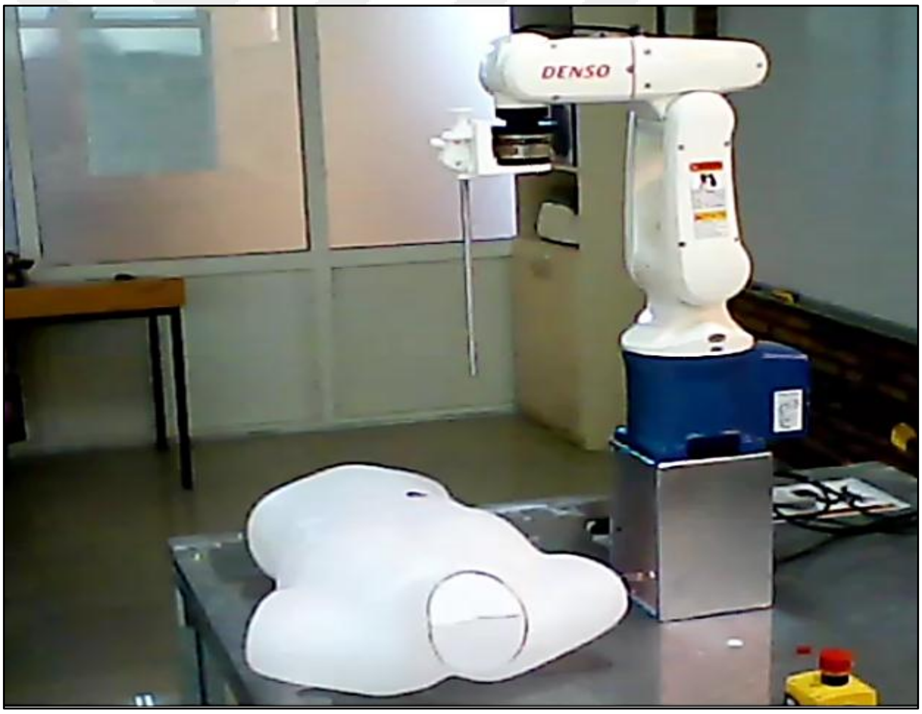


Figure 4.22 Robotic arm with Thoracoscope, and human body model

4.5.2 Robotic Surgery; Kinematics

Robotic-assisted minimally invasive surgery (RAMIS) is implemented by robotic manipulator. Thoracoscope tool is attached to robot and penetrated in the skin through a trocar. To avoid damaging the patients' tissues at the insertion point, the lateral displacements at this point is kept to a minimal. This adds a kinematic constraint to the manipulation, and is referred to as the Remote Center of Motion (RCM) mechanism. In a remote center of motion (RCM), a small incision fulcrum point (trocar point) in the patient's body surface gives the surgeon possibility to reach positions inside the patient without creating an invasive cut. In most of the surgical operations, three or more incisions are needed, two for the instruments and one for the Thoracoscopic camera. In this study, Thoracoscopic camera is attached to six DOF Denso robotic arm. The RCM is applied to control the Thoracoscope tip in RAMIS. The allowed movement of the shaft of Thoracoscope is restricted to one translational insertion motion, and three rotations about the corresponding point (F). Therefore, the shaft of the Thoracoscopic camera has only four DOF. Figure 4.23 shows Thoracoscopic motion and trocar insertion point (F).

The control coordinate for moving a surgical scope inside the patient's body would be Cartesian control. Therefore, the kinematics of surgical robot is expressed by a homogeneous transformation matrix from base to end effector (Thoracoscope tip). The trocar point is assumed to be an imaginary point (F) of the end effector, the transformation matrix from base to wrist is ${}^{0}_{G}T$, the translational motion from point F to Thoracoscope tip is ${}^{F}_{E}T$ which presents the inserted part inside patient's body. The transformation matrix from wrist to point F is ${}^{G}_{F}T$ which is the part outside patient's body from the Thoracoscope. Figure 4.24 shows the suggested notation for RCM kinematics.

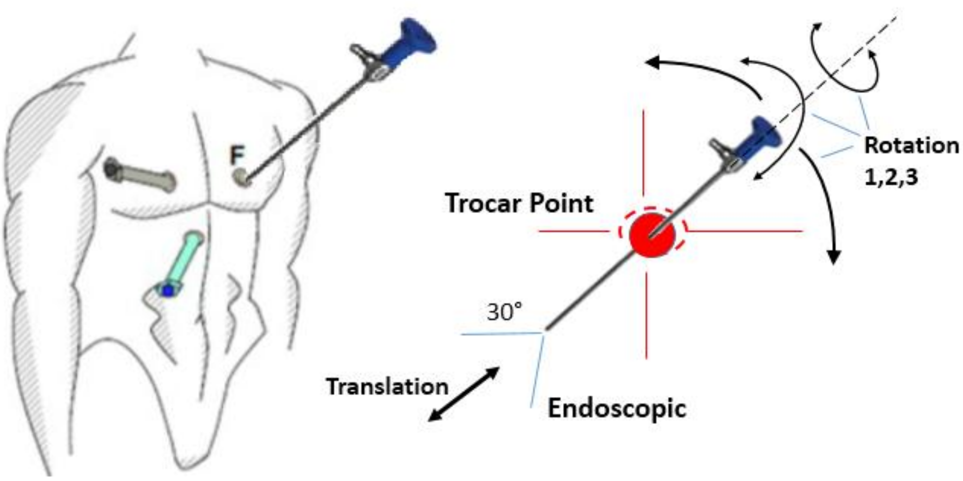


Figure 4.23 Important movement in RAMIS

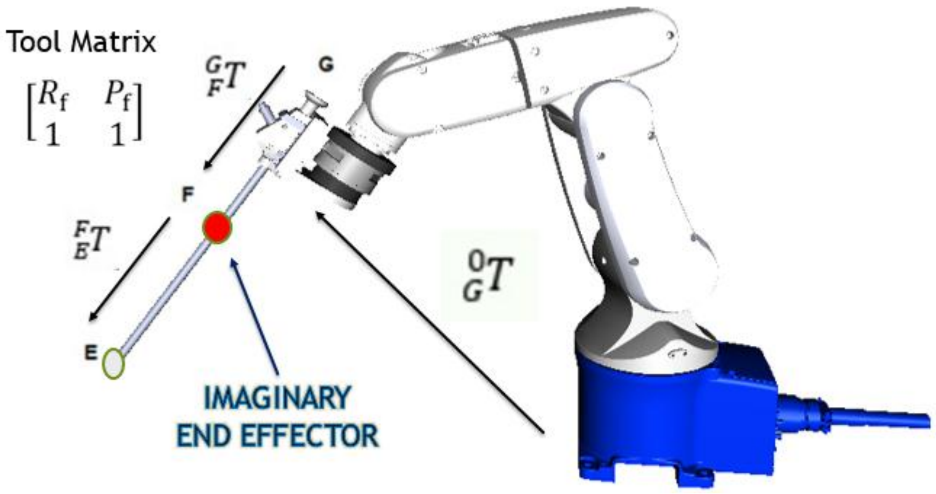


Figure 4.24 Suggested notation for RCM kinematics

A homogeneous transformation matrix (${}^{0}_{E}T$) presents the orientation and position of scope with respect to base of robot:

$$ {}^{0}_{E}T = \begin{bmatrix} {}^{0}_{E}R & {}^{0}_{E}P \\ 0 & 1 \end{bmatrix} \tag{4.28} $$

$$ {}^{0}_{E}T = {}^{0}_{G}T \; {}^{G}_{F}T \; {}^{F}_{E}T \tag{4.29} $$

Where ${}^{0}_{E}P$ is the position vector of Thoracoscope camera in workspace coordinate. It is given by surgeon through the input device connected with the control program. ${}^{0}_{E}R$ is the orientation of Thoracoscope. According to RCM technique, the centered point F has a fixed position and changed orientation and also translational motion. The kinematic control for this point is given as:

$$ {}^{0}_{F}T = {}^{0}_{G}T \; {}^{G}_{F}T \tag{4.30} $$

$$ {}^{0}_{F}T = \begin{bmatrix} {}^{0}_{F}R & {}^{0}_{F}P \\ 0 & 1 \end{bmatrix} \tag{4.31} $$

$$ {}^{G}_{F}T = \begin{bmatrix} R_{fg} & P_{fg} \\ 1 & 1 \end{bmatrix} \tag{4.32} $$

$$R_{fg} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.33)$$

$$P_{fg} = \begin{bmatrix} P_{fg\_x} \\ P_{fg\_y} \\ P_{fg\_z} \end{bmatrix} \quad (4.34)$$

P_{fg} is the position vector of trocar point from the wrist of the robot in wrist coordinate. P_{fg_x} is the Thoracoscope length in x-axis. P_{fg_y} is the Thoracoscope length in the y-axis, it is equal to 0.067m, while P_{fg_z} is the distance of trocar point from the wrist in the z-axis, and it is equal to subtraction of inserted part ($P_{insertion}$) from total length of Thoracoscope (0.36m).

$$P_{fg\_z} = 0.36 - P_{insertion} \quad (4.35)$$

The technique manipulates Thoracoscopic camera by assuming the imaginary end effector point. It needs the forward and inverse kinematics analysis for the robotic system. The position and orientation of Thoracoscopic tip are obtained by using forward kinematics as follows:

$$_{F}^{0}T = ForwardKinematics\,[q] \quad (4.36)$$

The joint angles are obtained by inverse kinematics for desired position / orientation of point F and insertion $P_{insertion}$.

$$[q] = InverseKinematics\,(_{F}^{0}T) \quad (4.37)$$

The position of trocar point is set in operation initialization while the variables of orientation and insertion are required in inverse kinematics solution of RCM.

$$[q] = InverseKinematics \begin{pmatrix} \theta_x \\ \theta_y \\ \theta_z \\ P_{insertion} \end{pmatrix} \quad (4.38)$$

SimMechanics model is interacted with a Virtual Reality Model representing RTS system. The robot is visualized in Virtual Reality Model (VRM). Figure 4.25 shows a Virtual Reality Model for robotic Thoracoscopic surgery.

Figure 4.25 VRM for robotic Thoracoscopic surgery

4.5.3 Robotic Surgery; Proposed Control Method

The proposed haptic control method is modified with joystick control guide and real time reconfiguration for RCM kinematics. The robotic system is controlled by two ways; HRI and joystick guide. In initialization of surgery, the operator can put the system in appropriate position and move the Thoracoscopic tool to the trocar point on patient skin by using direct HRI. While during surgical operation, the surgeon can control the system by using joystick control guide, so that, the trocar position (P_F), trocar orientation R_F and insertion length P_E can be changed. These parameters definitely reconfigure the kinematics of robotic surgery (RCM). Figure 4.26 shows the proposed control method for robotic surgery.

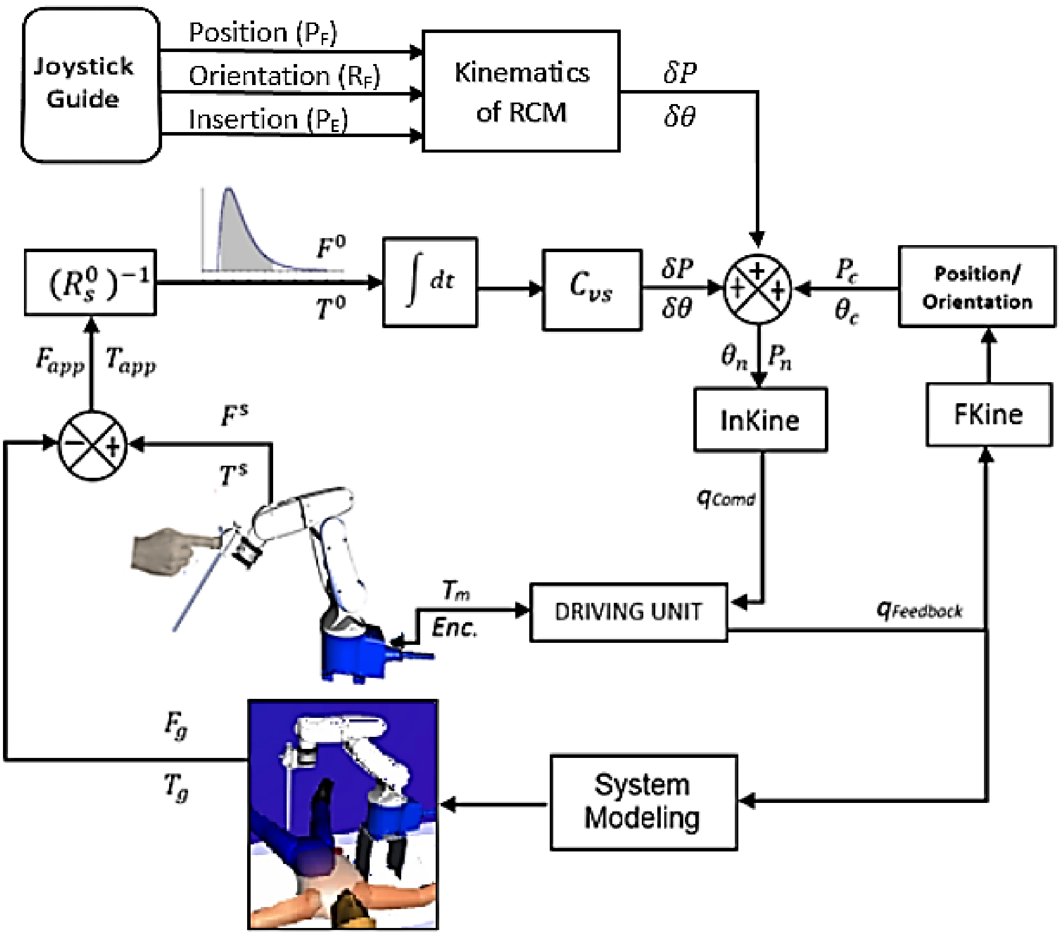


Figure 4.26 Proposed control method for robotic surgery

CHAPTER 5

EXPERIMENTS AND APPLICATIONS

5.1 Experimental Verification for System Model

The signals of force and torque are dependent variables in haptic controller. The system model must have a similar F/T response for the gravitational effect of gripper. The experiment of system model verification is carried out by applying a free motion in joint space coordinates without any external applied force. Motion of robot is applied by using a step input change to the joint angles one by one. The force and torque feedback signals are measured. The experiment is evaluated for the system model F/T response.

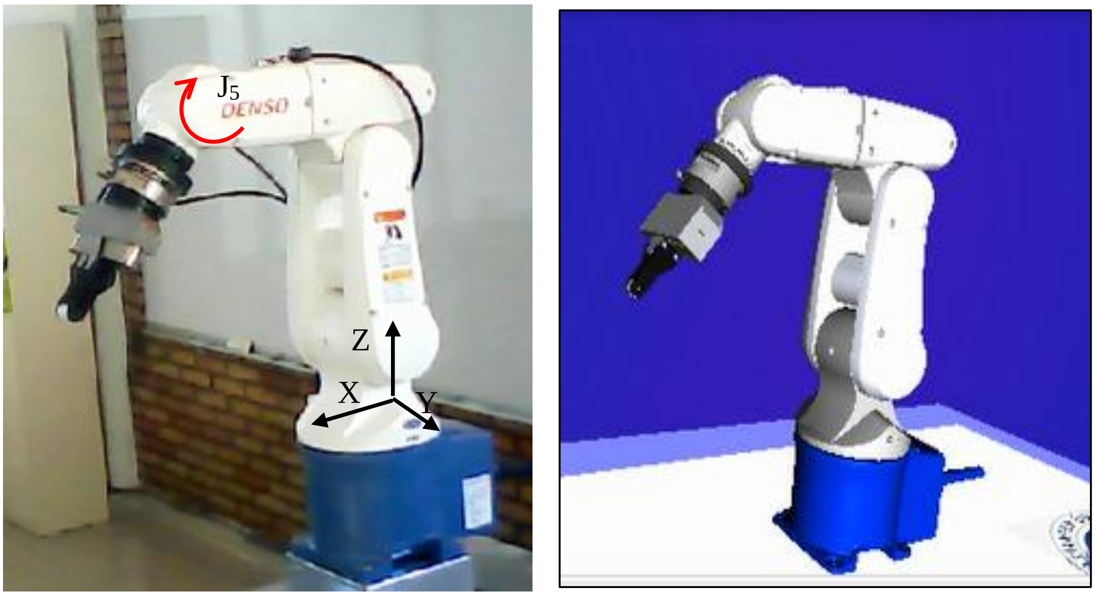
5.1.1 Force Response Verification

Both systems (real system and system model) are set in home position. The signal of force feedback is nulled to have initially zero feedback at home position. The joint angles are slightly changed. The force signal is measured for both systems. Table 5.1 shows steps for the joint angles changing system model force verification. Figure 5.1 shows Step 2 which is changing the angle of joint 5 from -90º to -50º. This motion moves the gripper in x-z plane. In this step, the force feedback must have a signal response in x and z directions. Figure 5.2 shows Step 3 changing the angle of joint 1 (J1 from 0º to 40º). This motion rotates the system about z-axis and has gripper movement in x-axis and y-axis. The force feedback must show an additional signal response in y directions. Figure 5.3 shows Step 4 changing the angle of joint 6 from 0º to 40º. This rotation about z-axis of gripper coordinate causes a force feedback varying by orientation change. Figure 5.4 shows Step 5 changing the angle of joint 3 from -90º to -75º.

In this step, the force feedback must have a jump of signal response in z direction. Steps 6 - Step 9 repeat the system in home position and check the zero force feedback signal at this position. Figure 5.5 shows the completed path generation from this experiment, and projected onto x-z plane, y-z plane, and x-y plane.

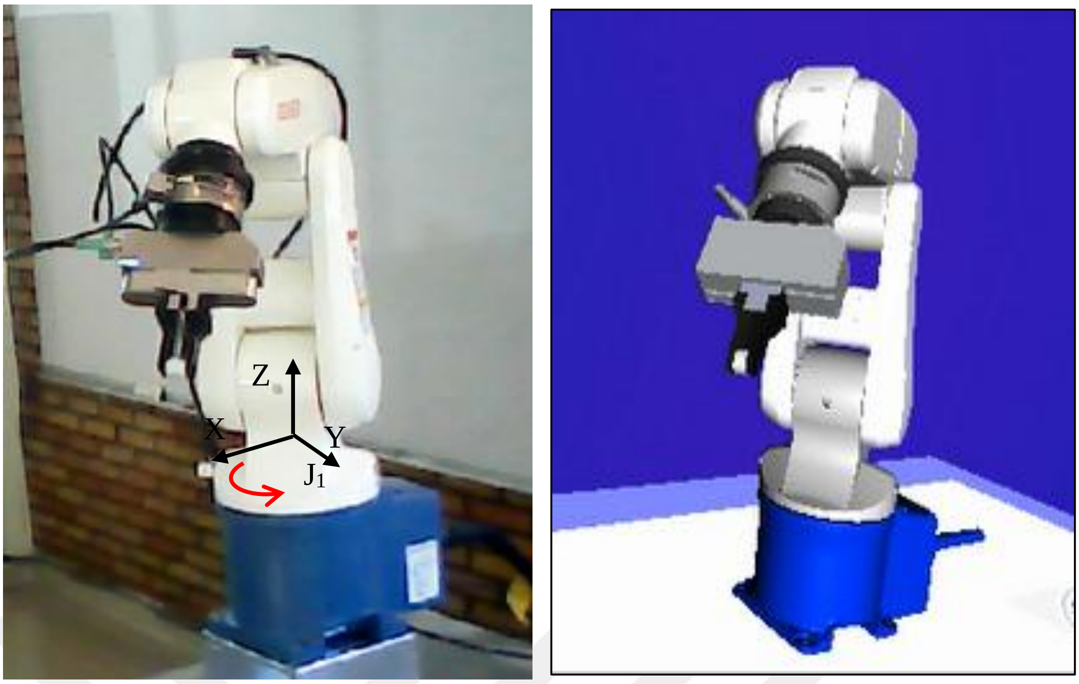
Table 5.1 Steps of joint angles (Degree) system model force verification

Steps	Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6
1 (Home)	0	0	-90	0	-90	0
2	0	0	-90	0	-50	0
3	40	0	-90	0	-50	0
4	40	0	-90	0	-50	40
5	40	0	-75	0	-50	40
6	40	0	-90	0	-50	40
7	40	0	-90	0	-50	0
8	0	0	-90	0	-50	0
9 (Home)	0	0	-90	0	-90	0



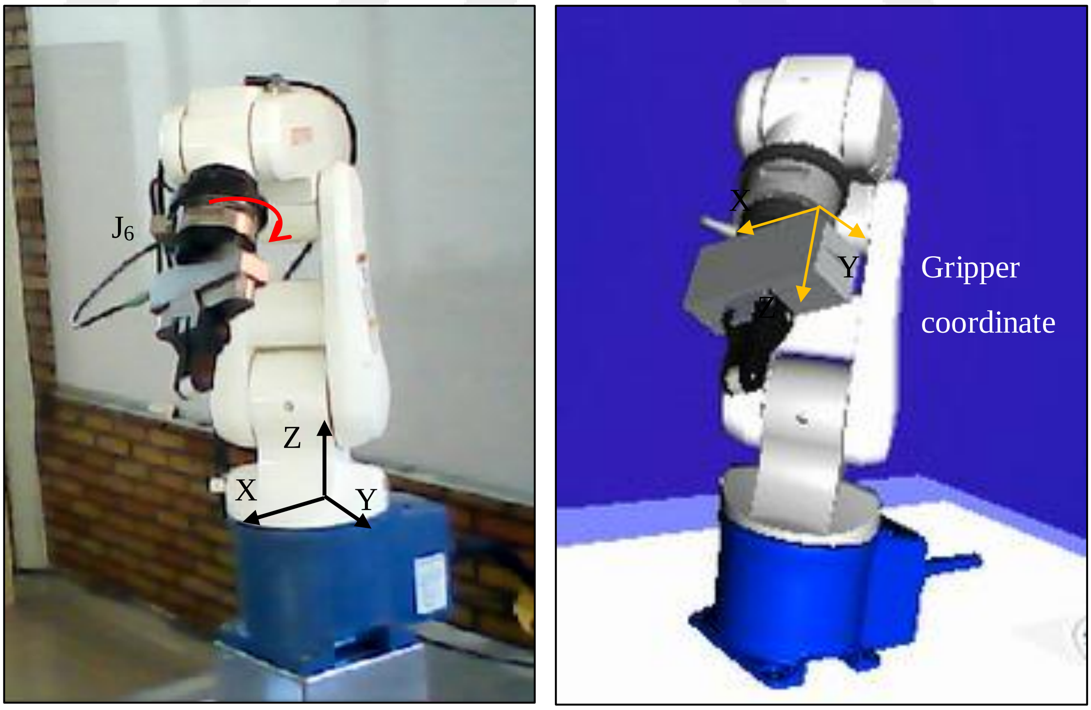
(a) (b)

Figure 5.1 Force response verification, Step 2 changing in the angle of joint 5. (a) Real system in position of Step 2. (b) System model in position of Step 2.



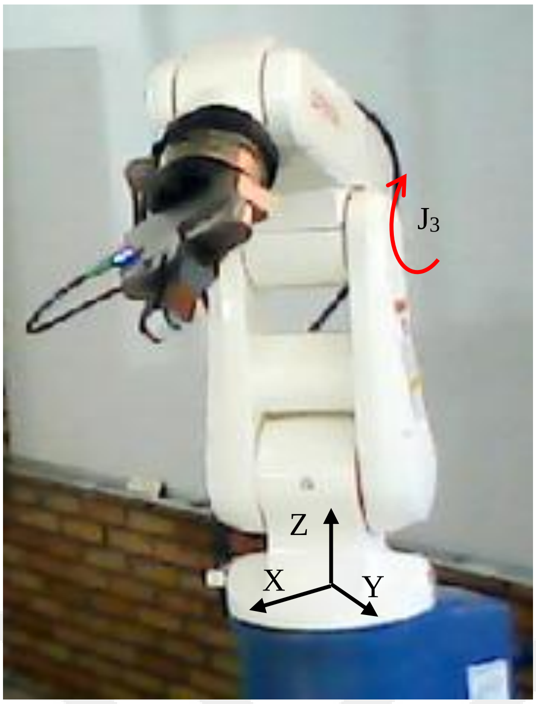
(a) (b)

Figure 5.2 Force response verification, Steps 3 changing the angle of joint 1. (a) Real system in position of Step 3. (b) System model in position of Step 3

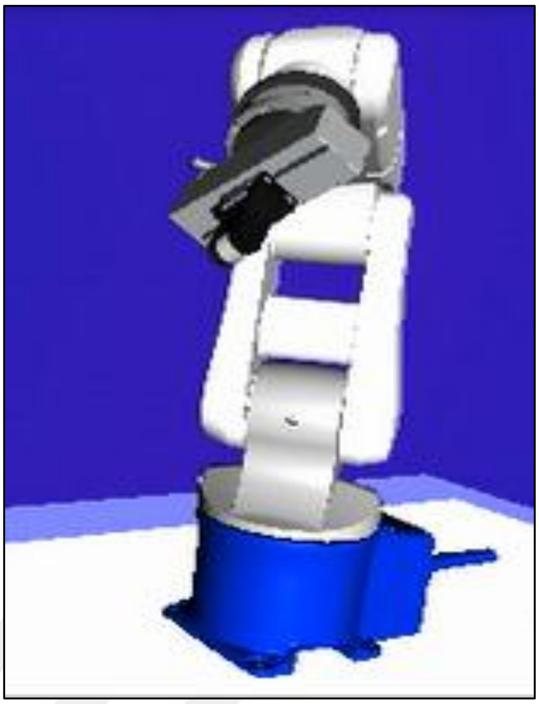


(a) (b)

Figure 5.3 Force response verification, Steps 4 changing the angle of joint 6. (a) Real system in position of Step 4. (b) System model in position of Step 4

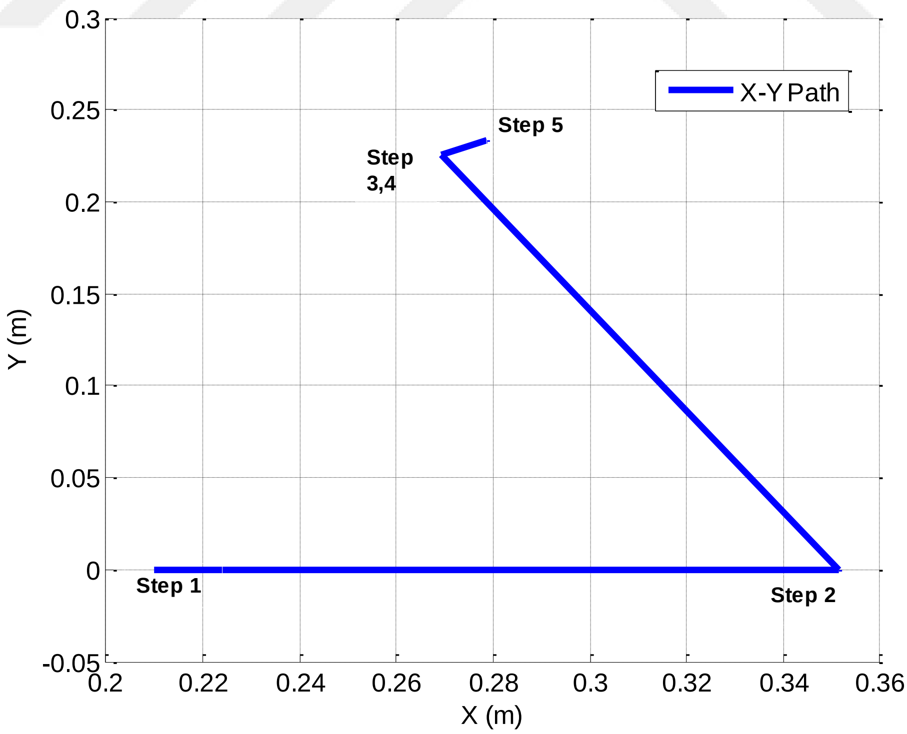


(a)

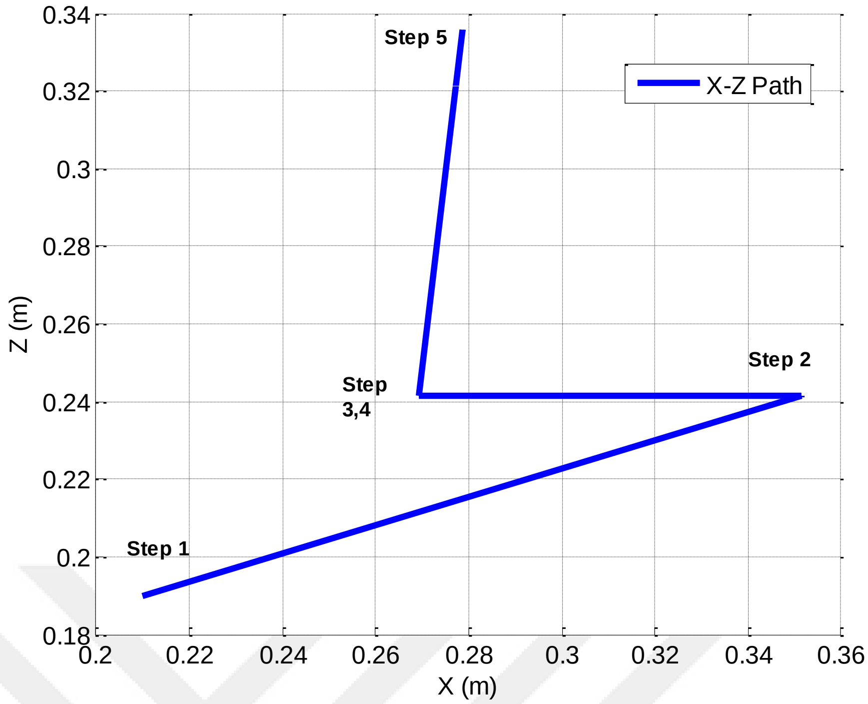


(b)

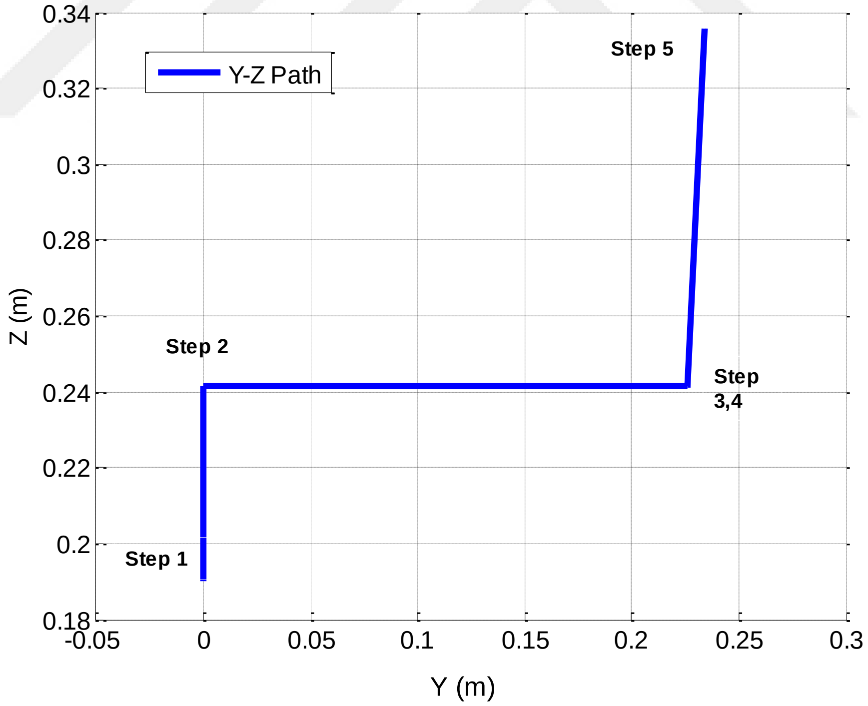
Figure 5.4 Force response verification, Step 5 changing in the angle of joint 3. (a) Real system in position of Step 5. (b) System model in position of Step 5



(a)



(b)



(c)

Figure 5.5 Completed applied path for system model force feedback experimental verification. (a) Projected path onto x-z plane, (b) Projected path onto y-z plane, and (c) Projected path onto x-y plane

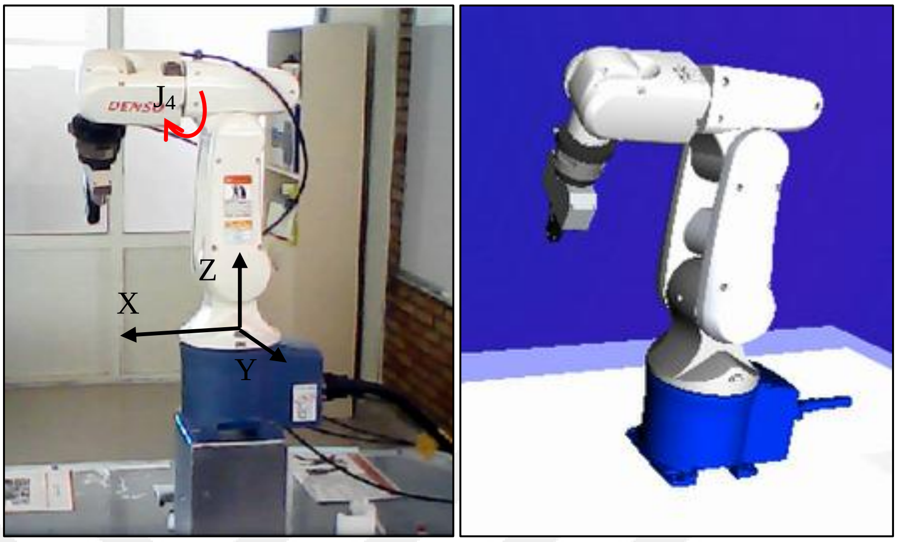
5.1.2 Torque Response Verification

The experiment of system model torque feedback verification is carried out by applying a free motion in joint space coordinates. Motion of robot is changed by varying the joint angles one by one. The torque feedback signals are measured. Both systems (real system and system model) are set in home position. The signal of torque feedback is nulled to have initially zero feedback at home position. The joint angles are slightly changed and the torque signal is measured for both systems. Table 5.2 shows steps for joint angles changing in system model torque verification. Figure 5.6 shows Step 2 which is the angle of joint 4 from 0º to -30º. This motion rotates the gripper counter clockwise (ccw) about x-axis. In this step, the torque feedback Tx must have a positive signal response from movement of initial of gripper. Figure 5.7 shows the Step 3 changed in the angle of joint 4 from -30º to 45º. This motion rotates the gripper clockwise (cw) about x-axis. The torque feedback Tx must have a negative signal response. In Step 4 the joint angle 4 is changed to zero. Figure 5.8 shows Step 5 changed in the angle of joint 5 from -90º to -30º. This motion rotates the gripper counter clockwise (ccw) about y-axis. The torque feedback Ty must have a negative signal response in this step. Figure 5.9 shows Step 7 changed in the angle of joint 6 from 0º to -35º. This motion rotates the gripper clockwise about z-axis. In Step 8 the joint angle 6 is changed to 20º.

Table 5.2 Steps for joint angles changing system model torque verification

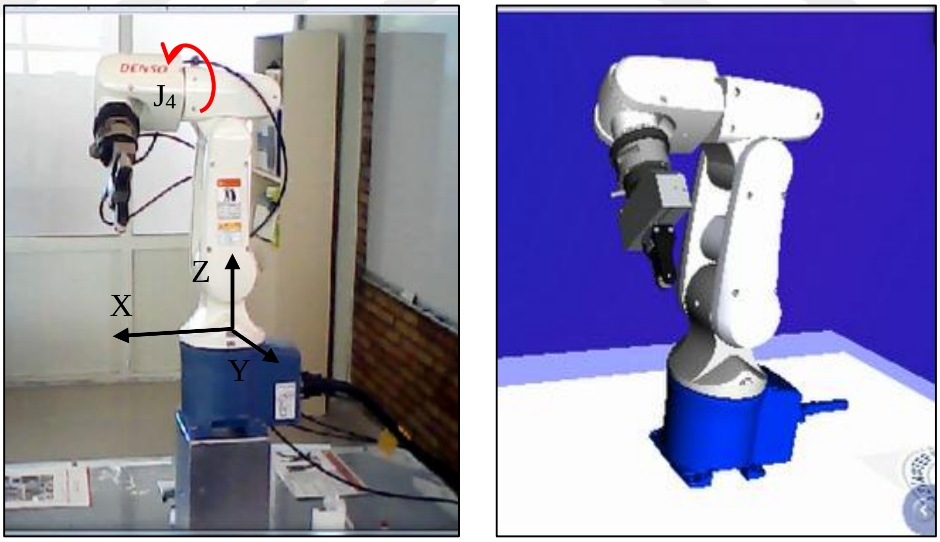
Steps	**Joint 1**	**Joint 2**	**Joint 3**	**Joint 4**	**Joint 5**	**Joint 6**
1 (Home)	0	0	-90	0	-90	0
2	0	0	-90	-30	-90	0
3	0	0	-90	45	-90	0
4	0	0	-90	0	-90	0
5	0	0	-30	0	-30	0
6	0	0	-90	0	-90	0
7	0	0	-90	0	-90	-35
8	0	0	-90	0	-90	20
9 (Home)	0	0	-90	0	-90	0

Last Step is to put both systems at home position. Figures 5.10 shows the completed path generation from this experiment, and it is projected onto x-y plane.



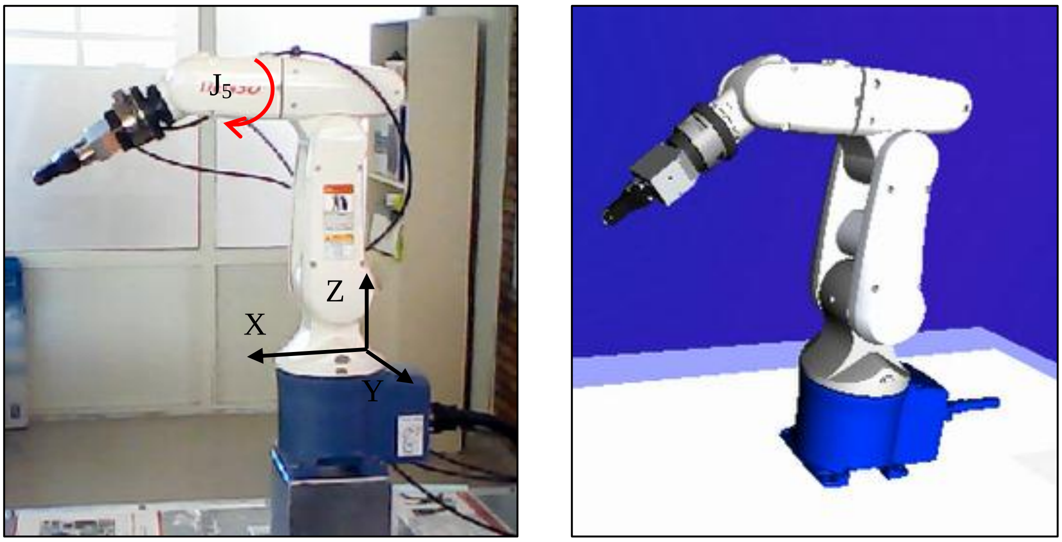
(a) (b)

Figure 5.6 Torque response verification, Step 2 changing the angle of joint 4. (a) Real system in position of Step 2. (b) System model in position of Step 2



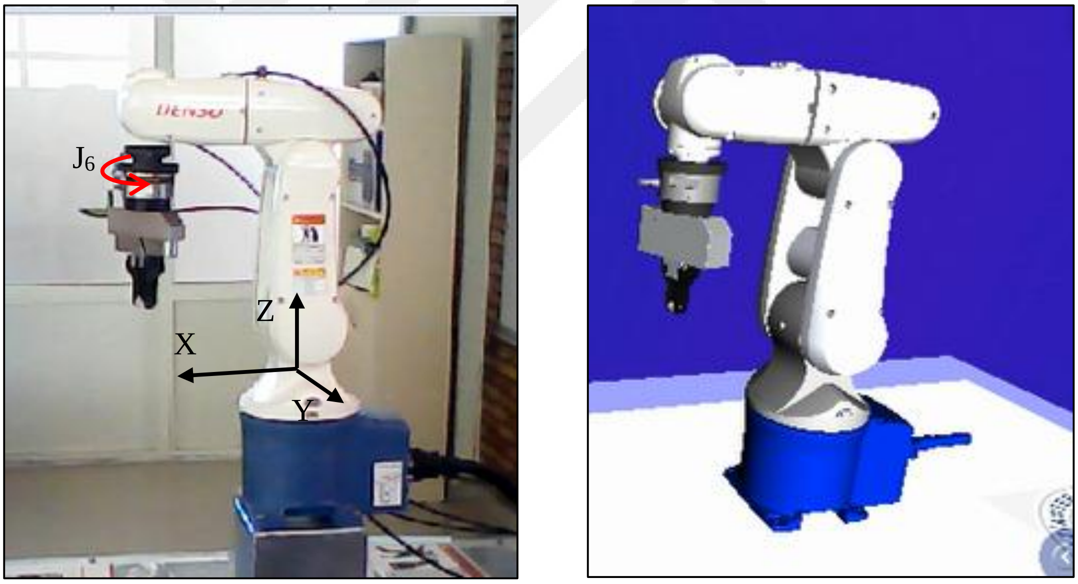
(a) (b)

Figure 5.7 Torque response verification, Step 3 changing the angle of joint 4. (a) Real system in position of Step 3. (b) System model in position of Step 3



(a) (b)

Figure 5.8 Torque response verification, Step 5 changing the angle of joint 5. (a) Real system in position of Step 5. (b) System model in position of Step 5



(a) (b)

Figure 5.9 Torque response verification, Step 7 of changing in the angle of joint 6. (a) Real system in position of Step 7. (b) System model in position of Step 7

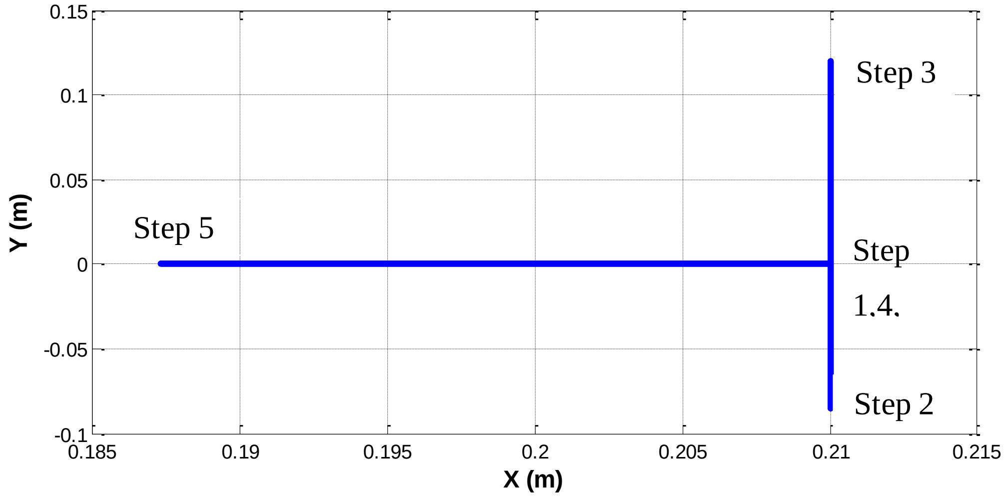


Figure 5.10 Applied path projected onto x-y plane for system model torque feedback experimental verification

5.2 Experimental Verification for Proposed Control Methods

A verification is performed for the proposed control method to realize the system response for external force and torque in HRI environment. The effect of gravity of gripper is eliminated, so that only the external force/torque will drive the robot directly. The experiment set up is done to put the robot in home position. A force in different directions is applied on the end effector. In this experiment, the operator applies a force then the robot moves in a direction relative to the applied force. Haptic experiments are carried out on the system robot as follows:

1) The end effector is pulled forward in x direction by hand, Figure 5.11 shows positive direction x-axis movement for the robot. The effective force Fx applied by operator. It is dominated to control the motion of robot in the x-axis.
2) The end effector is pushed left in y direction by hand, Figure 5.12 shows negative direction y-axis movement for the robot. The effective force Fy applied by operator. It is dominated to control the motion of robot in the y-axis.
3) The end effector is pulled down in z direction by hand, Figure 5.13 shows down direction movement for the robot. The effective force Fz is applied by operator. It is dominated to control the motion of robot in the z-axis.

Figure 5.11 Positive direction x-axis movement for the robot by HRI

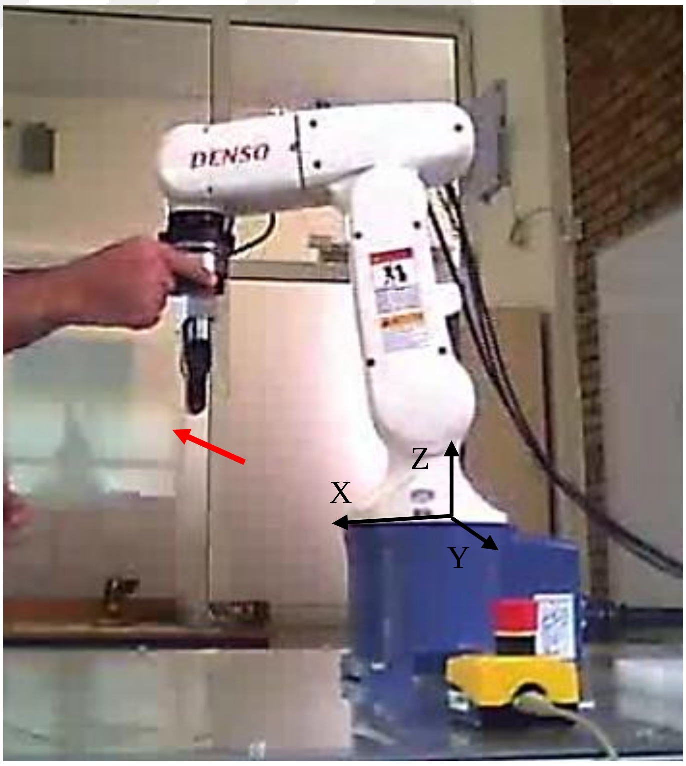


Figure 5.12 Negative direction y-axis movement for the robot by HRI

Figure 5.13 Down direction z-axis movement for the robot by HRI

4) The experiment is also applied to verify the torque controlling method. The end effector is rotated about y-axis by hand, Figure 5.14 shows rotation the robot. The effective torque Ty applied by operator is dominated to control the orientation of robot in the y-axis.
5) The end effector is rotated about x-axis by hand. Figure 5.15 shows rotation of the robot, the effective torque Tx applied by the operator is dominated to control the orientation of robot in the x-axis.
6) The end effector is rotated about z-axis by hand, Figure 5.16 shows rotation of the robot. The effective torque Tz applied by the operator is dominated to control the orientation of robot in the z-axis.

Figure 5.14 Rotation about y-axis for the robot by HRI

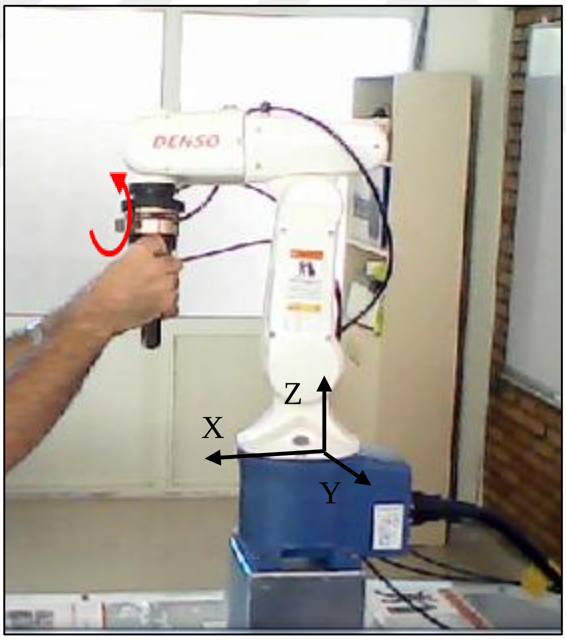


Figure 5.15 Rotation about x-axis for the robot by HRI

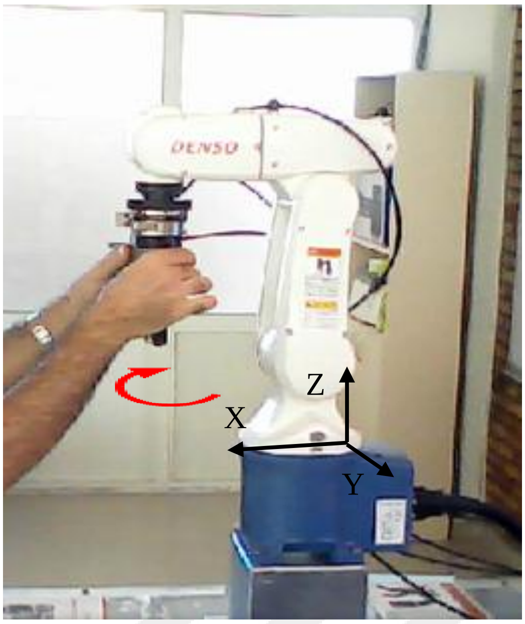


Figure 5.16 Rotation about z-axis for the robot by HRI

5.3 Application of Online Motion Learning

Robot motion learning is implemented by online teaching and robot task execution. The experiment is to move the end effector along a nonlinear curve in 3D workspace by direct operator's hand. The desired path (taught path) and the repeated path are generated. Robot learning is achieved by three different path repeating variables (joint angle, joint velocity, and F/T signal), the taught path variables are recorded with time. They are reapplied individually in executing task operation. In the first case of study, the motion is taught by recording joint angle. The second case is the motion learning by recording joint velocity. The third case is the motion learning by recording external F/T signal. The recorded path is reapplied on the robot in task execution time. The robot is demonstrated a continuous smooth motion. Suggested path is repeated autonomously by the system. The desired path (taught path) and the repeated path are illustrated in Figure 5.17.

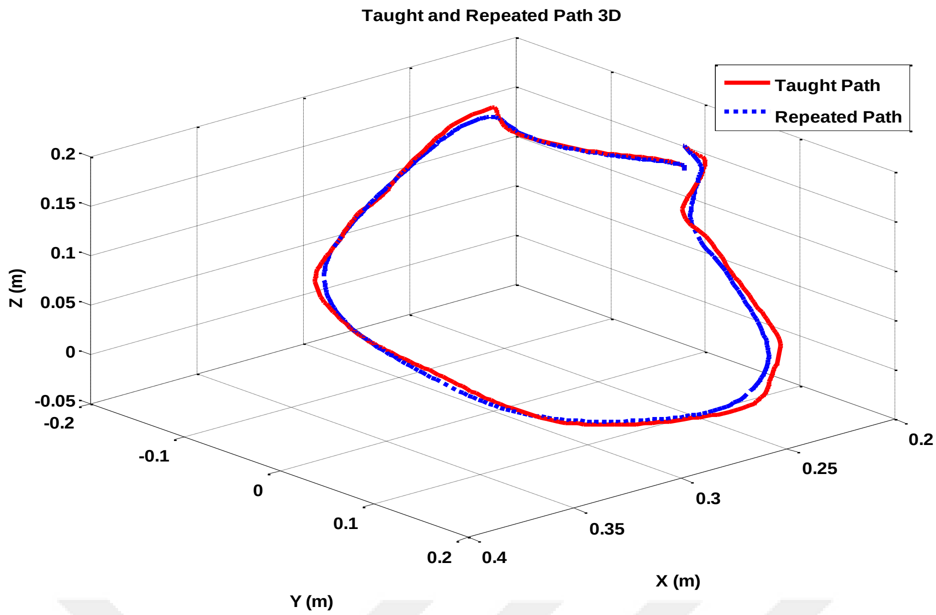


Figure 5.17 The desired path (taught path) and the repeated path in 3D workspace for the application of online motion learning

5.4 Experiment on ANN of inverse kinematics solution

Having completed the training of proposed ANN and traditional ANN, the experiments of movement are carried out by the robotic system. The gripper motion is in a helical path within the workspace area. The input path for gripper tip is generated by $P_x(k) = c_1 \sin(k)$, $P_y(k) = c_2 \cos(k)$ and $P_z(k) = c_3 k$. In this experiment, the robot has to follow a sinusoidal rotation in x-axis and y-axis while the motion in z-axis is a linear path. The constants c_1 and c_1 are the amplitude of radius on the x and y axes respectively, the constant c_3 is the pitch of a helix, and k is the sample time. Figure 5.18 shows the real robotic system and the virtual model to demonstrate helical motion path by ANN. The inverse kinematics solution for robot motion is achieved by proposed ANN and traditional ANN.

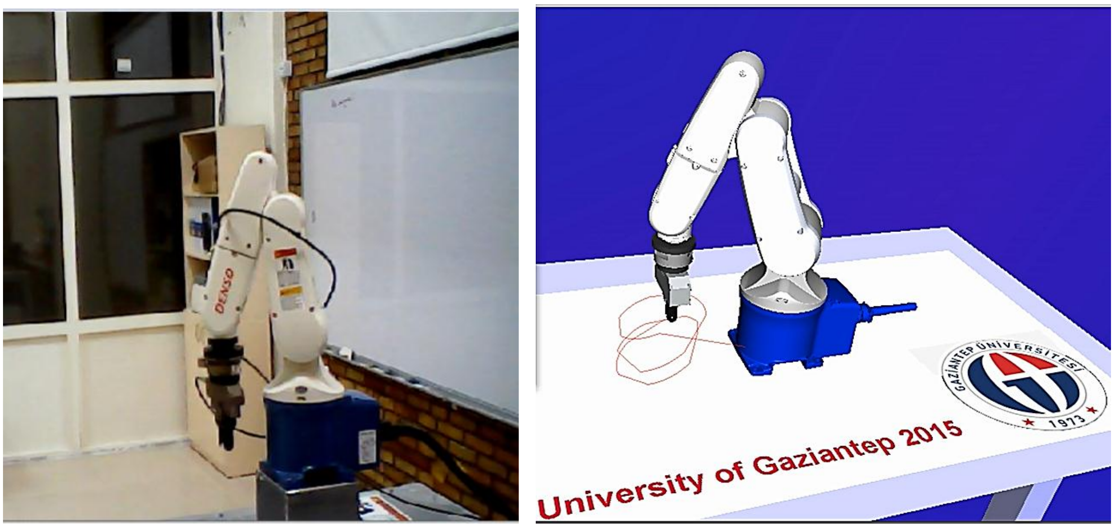


Figure 5.18 Experiment on robotic motion by ANN
(Movement of end effector through helical path)

The movements of robot are executed in circular and linear motion. The pitch of path of the end effector is changed by varying the position in z direction relative to workspace coordinates. The rotation of end effector is performed by changing the position to x and y directions. Two paths are generated by using proposed ANN, and traditional ANN with desired path. Figure 5.19 illustrates the paths configuration of the end effector by the robot movement.

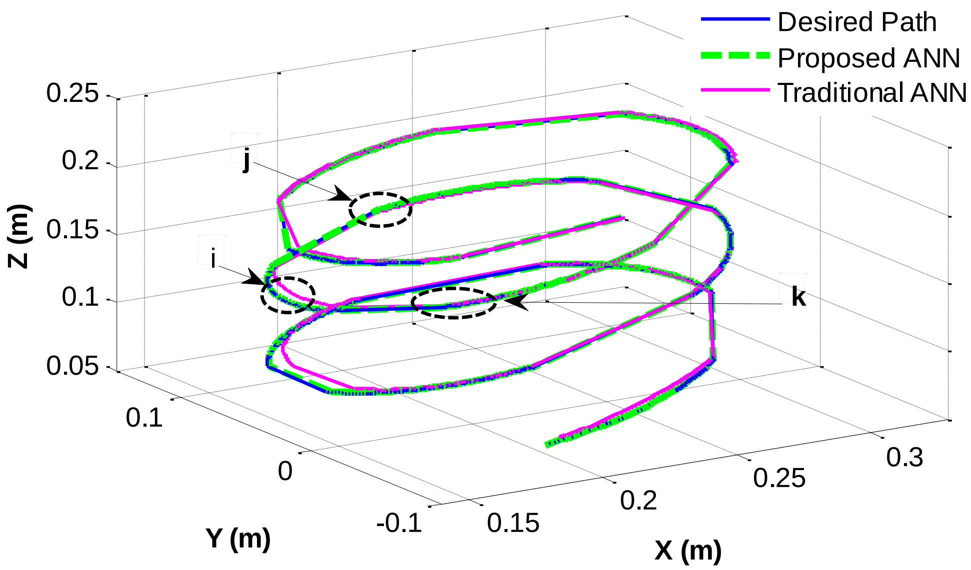
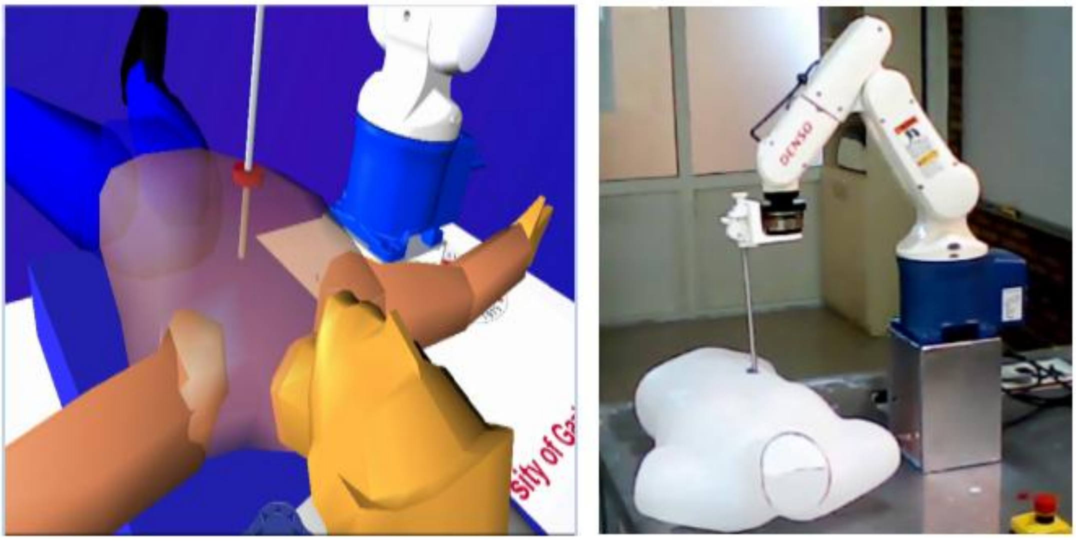


Figure 5.19 Paths configuration generated by robot in two different ANNs

5.5 Application of Robotic Assisted Thoracoscopic Surgery (RATS)

The surgical Thoracoscopic manipulation are carried by the robotic system. The Thoracoscope is inserted through a small trocar point and the motion is in a circular path inside a plastic human body model. The desired path for endoscope tip is applied by equations $\theta_x(k) = c_1 \sin(\omega k)$ and $\theta_y(k) = c_2 \cos(\omega k)$. In this experiment, the incision point has to follow a sinusoidal rotation in x-axis and y-axis while the endoscope tip has to move over a circular path, the constants c_1 and c_1 are the amplitude of radius on the x and y-axes respectively, the signal frequency is ω, and k is the number of samples. Figure 5.20 shows the real robotic system and the virtual model to demonstrate a robotic surgery endoscope manipulation. The kinematics controller for the remote centered of motion RCM is achieved by proposed controller design for RCM.

The movements for RCM are applied to linear insertion and rotational motion on trocar point. The depth of inserted part of the endoscope is changed by varying the position of imaginary fulcrum point on z direction relative to tool coordinates. The rotation of endoscope over trocar point is performed by changing the orientation (Rx, Ry, Rz) for the tool. The tip of the endoscope is moved in over the desired path. Figure 5.21 illustrates the path configuration of the endoscope by the robot.



(a)

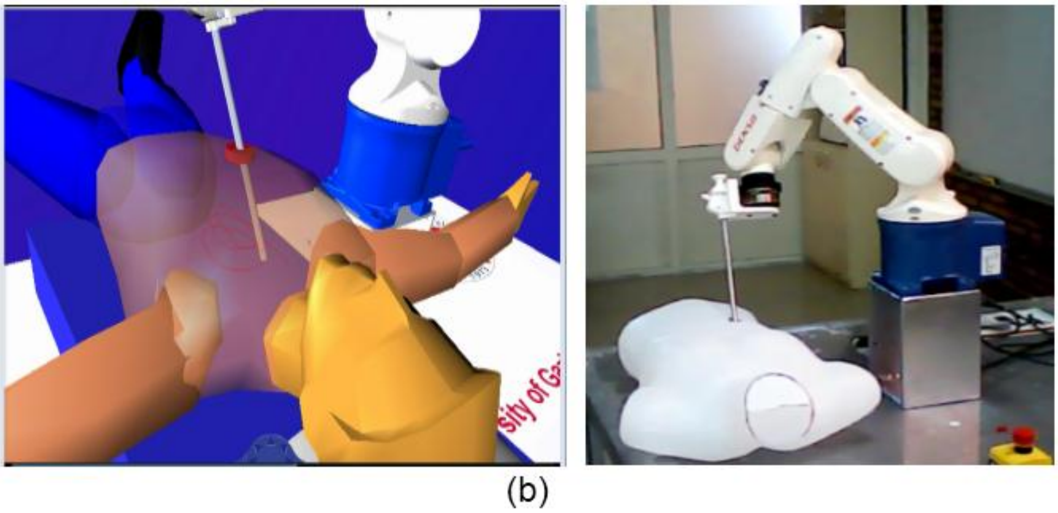


Figure 5.20 Experiment on robotic surgery Thoracoscope demonstration,
(a) Insertion of the Thoracoscope in human body model, (b) Moving of endoscope in a circular path through trocar point.

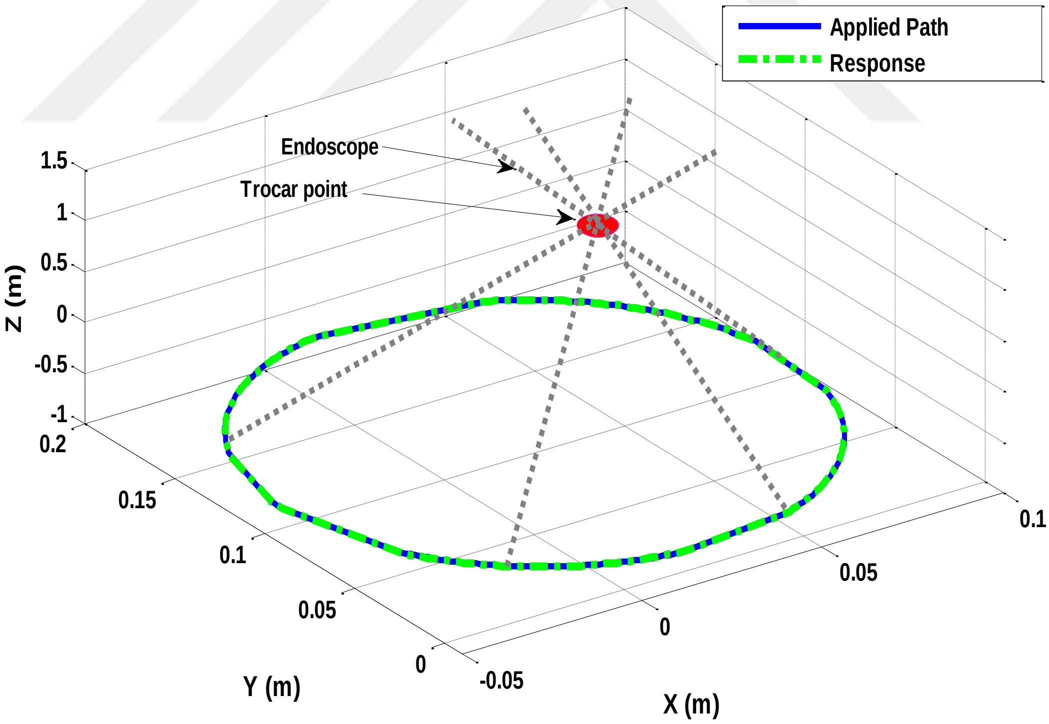


Figure 5.21 Path configuration of endoscope response by the proposed control method

CHAPTER 6

RESULTS AND DISCUSSION

6.1 Results of System Model Force Response Verification

The output signals from system model verification in Chapter 5 (Section 5.1.1) are analyzed here. In free motion, both F/T sensors on the real system and on the system model are responded to the gravitational load of gripper. The resultant forces from both systems are collected. Table 6.1 illustrates the forces for both of real system and system model. Figure 6.1 shows the forces feedback with time during the experiment of force response verification. Initially the feedback signal has force in z-axis equal to the weight of gripper. This force is initially converted to zero of the first step of Table 6.1, which refers to system homing. In Step 2 of movement, the angle of joint 5 is set to -50º. The system is moved forward in both x-axis direction and upward in z-axis direction. The F/T sensor measured the resultant forces from gripper inertia for real system (Denso robot) is Fx = -10N, Fz = -8N. For the system model (SimM) Fx = -10.5N, Fz = -8.5N. The movement in x-axis is in positive direction so that the inertial force in x-axis is in negative value. The same thing is happened for the force signal during the movement in z-axis.

Table 6.1 Feedback forces for both of real system and system model

Steps	Real System Force Feedbacks (N)			System Model Force Feedbacks (N)		
	Fx	**Fy**	**Fz**	**Fx**	**Fy**	**Fz**
1	0	0	0	0	0	0
2	-10	0	-8	-10.5	0	-8.5
3	-9	-3	-7	-9	-4	-7
4	-8	-0.5	-5	-8.5	-1	-5.5
5	-10	-2	-8.2	-10	-3	-8

In Step 3 of movement, the angle of joint 1 is set to 40º. The system is moved in both x-axis negative direction and in y-axis positive direction. The F/T sensor produced a reduction is the resultant forces from gripper inertia for real system; Fx = -9N, Fz = -7N. For the system model Fx = -9N, Fz = -7N. The movement in x-axis is in negative direction so that the inertial force in x-axis is reduced. The F/T sensor produced a force feedback signal in y-axis which for real system; Fy = -3N,   and for the system model Fy= -4N. In step 4 of movement, the angle of joint 6 is set to 40º, the system is not moved in any axis. It had only change in orientation of gripper. The F/T sensor gave a visible change in force signals over each axis. The resultant forces form gripper inertia for real system; Fx = -8N, Fz = -5N, and Fy = -0.5N. For the system model Fx = -8.5N, Fz = -5.5N, and Fy= -1N. The movement in x-axis is in negative direction so that the inertial force in x-axis is reduced. The F/T sensor produce a force feedback signal in y-axis which is for the real system Fy = -3N, and for the system model Fy= -4N.

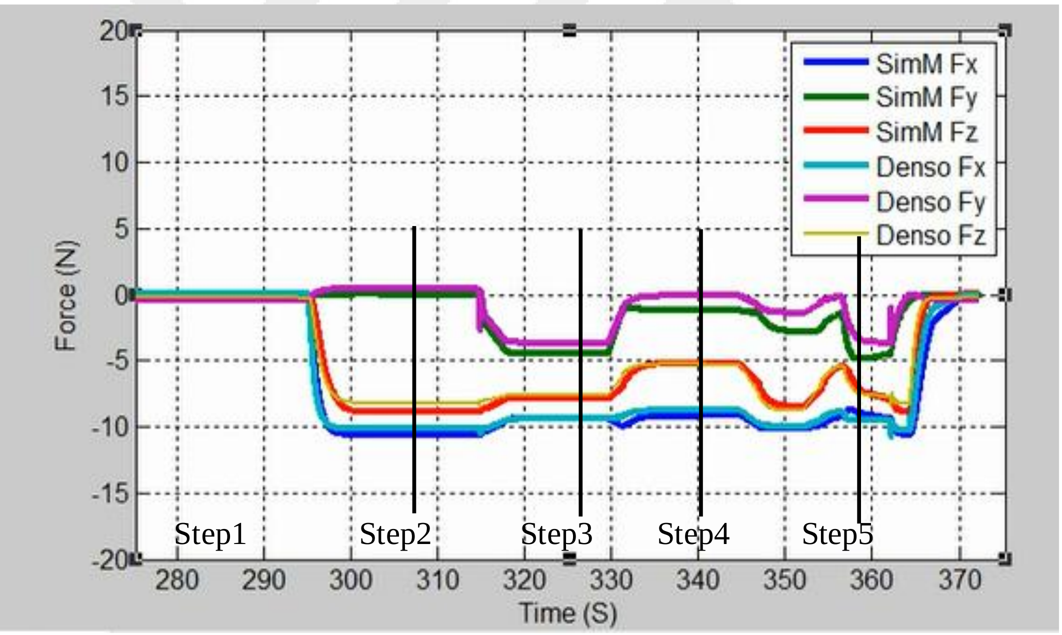


Figure 6.1 Force feedback for both real system, and system model in free motion

In step 5 the gripper coordinate is still rotated, and joint 3 is changed to 75º. Although that system is moved up in z-axis only, the forces in all axes are changed. The resultant forces form gripper inertia for the real system Fx = -10N, Fz = -8.2N, and Fy = -2N. For system model Fx = -10N, Fz = -8N, and Fy= -3N. This experiment proved that the real system and the system model have almost the same feedback signals. So the inertia effect of gripper cannot be neglected during the online HRI, and it is able to compensate by F/T sensor on the system model. The maximum force difference between real Denso system and the system model is less than 1 N (model error). The human hand exerts force ranging from 5 to 50N during the HRI process. So the system is not affected by this small difference.

6.2 Results of System Model Torque Response Verification

In the experiment of torque response verification, both systems; the real system and the system model are tested to the torque produced from the inertia of gripper. The resultant torque signal from both systems (F/T sensor) are collected. Table 6.2 shows torque signals with steps of experiment. Figure 6.2 illustrates torque feedback with time during the experiment of torque response verification. Initially the feedback signal is converted to zero in the first step of Table 5.2, which refers to system homing. In Step 2, the angle of joint 4 is set to -30º, the gripper is rotated about x-axis so the F/T sensor showed torque signal in x-axis. The torque feedback signal for real system is Tx = +1.9N.m, and for system model Tx= +1.85N.m. In this step the rotation is clockwise, so that torque in positive direction is caused by the inertia of gripper.

Table 6.2 Torque signals with steps of experiment for torque response verification

Steps	Real System Torque Feedbacks (N.m)			System Model Torque Feedbacks (N.m)		
	Tx	Ty	Tz	Tx	Ty	Tz
1	0	0	0	0	0	0
2	1.9	0	0	1.85	0	0
3	-2.2	0	0	-2.1	0	0
4	0	0	0	0	0	0
5	0	2.7	0	0	2.7	0
6	0	0	0	0	0	0
7	0	0	≈0	0	0	≈0
8	0	0	≈0	0	0	≈0

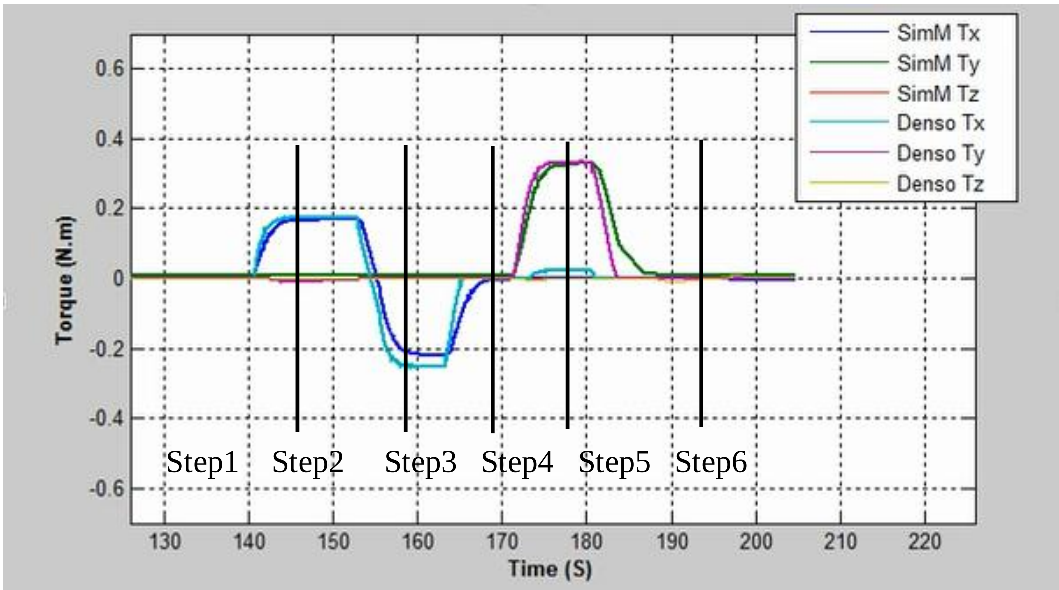
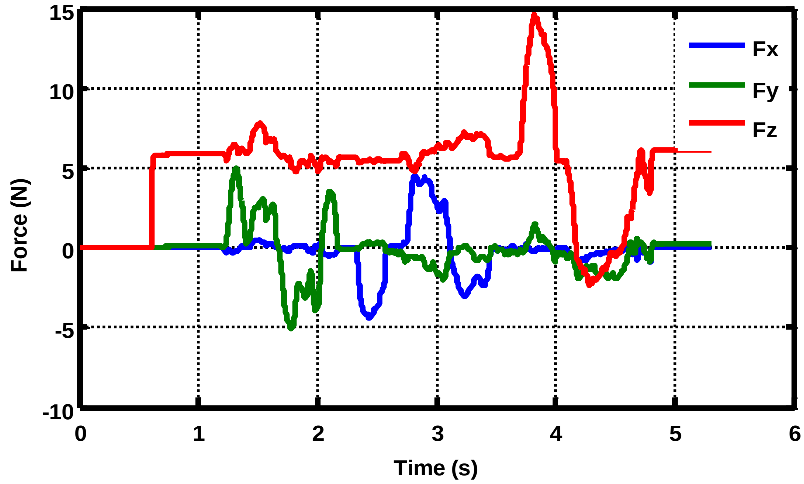


Figure 6.2 Torque feedback from real system and system model in free motion

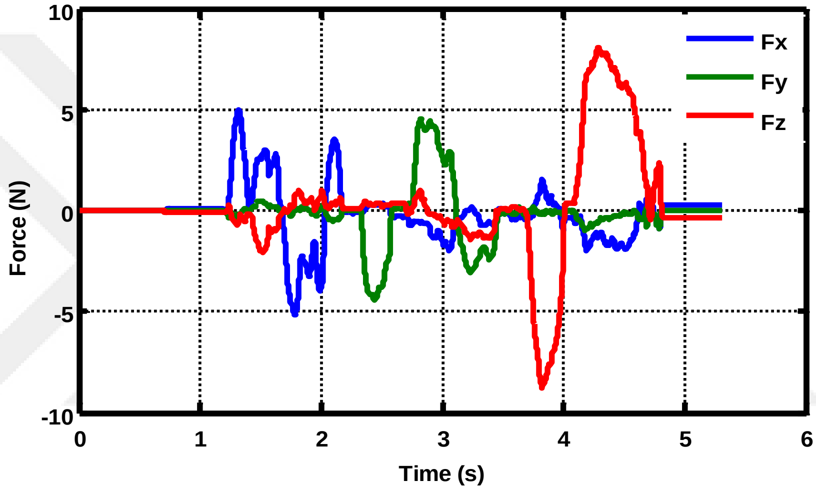
In Step 3, the angle of joint 3 is set to +45º, the gripper is rotated about x-axis so the F/T sensor showed torque signal in x-axis only. The torque feedback signal for real system is Tx = -2.2N.m, and for system model Tx= -2.1N.m. In this step, the rotation is counter clockwise so that inertia of gripper caused a torque in negative direction. In Step 4, the system is returned to home position, so that the torque signals became zero. In Step 5, the angle of joint 5 is set to -30º, the gripper is rotated about y-axis so the F/T sensor showed torque signal in y-axis only. The torque feedback signal for real system is Ty = 2.7N.m. The torque for system model Ty= 2.7N.m. In this step, the rotation is clockwise, so that inertia of gripper caused a torque in positive direction. In Step 6, the system is returned to home position so that the torque signals became zero. In steps 7, and 8, the angle of joint 6 is changed from -30º to +20º, the gripper is rotated about z-axis of gripper coordinate. In this step, the rotation is about the center of mass of gripper, so F/T sensor reads very low torque signal Fz from the inertia of gripper with slow rotational motion.

6.3 Results of Proposed Control Method (Force to Position)

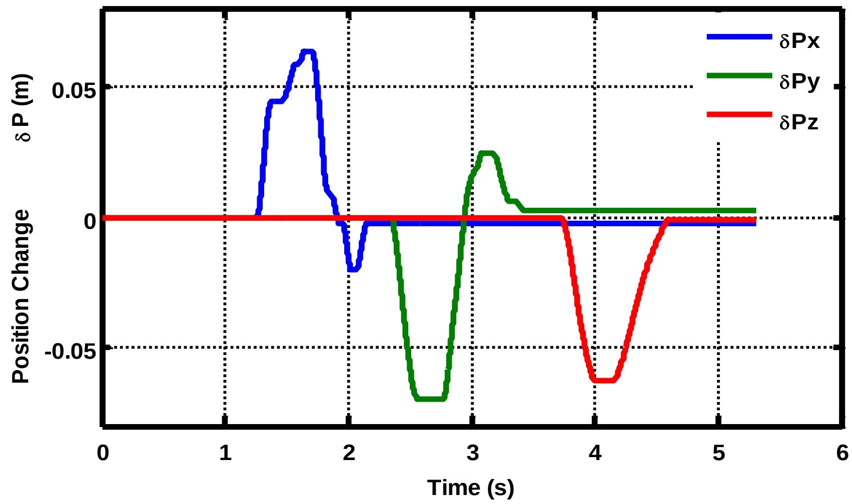
Verification is performed for the control method to realize the system response in HRI environment. The robot is set in home position. A force in different directions is applied on the end effector by human hand. Figure 6.3 shows force signal for F/P control method verification.



(a)



(b)



(c)

Figure 6.3 F/P verification. (a) Force signals in the sensor's joint coordinate, (b) Force signals in the robot's base coordinate, (c) Change in position δP

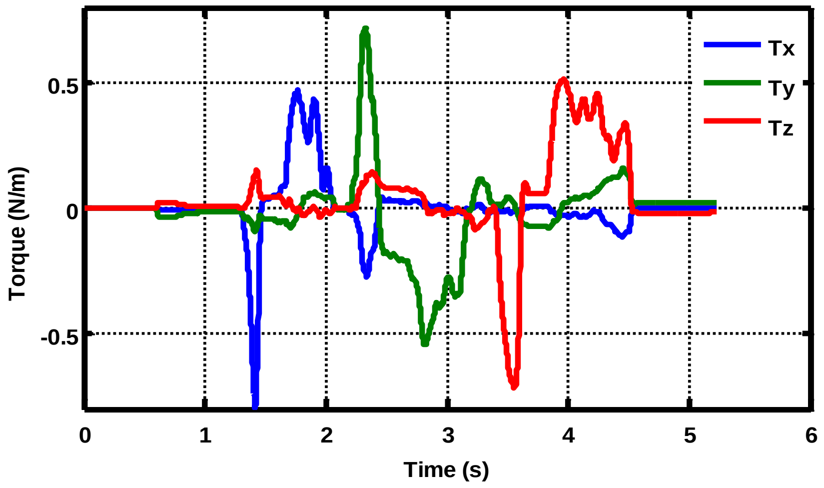
The robot is pulled in x direction by hand, and pushed in y direction then it is pulled down in z direction. Figure 6.3(a) shows the measured force F^s from F/T sensor. The first stage of the controller is to rotate the exerted force to frame coordinate. Then the force F^0 is obtained, as illustrated in Figure 6.3(b). The next stage in the controller is to find the changes in position δP. Which is shown in Figure 6.3(c). Referring to Figure 6.3(a), the force Fz is shifted up because of the weight of gripper, this weight load is removed from Fz signal by the compensation of system model. The rotation of forces from sensor joint coordinate to frame coordinate is clearly appeared in Figure 6.2(b). The Fx in joint sensor is transformed to the Fy in frame coordinate while the Fz is flipped, and this rotation in axes is due to the implementation of equation (4.13) in Chapter 4. The force F^0 is converted to the change in position δP by using the equation (4.16).

6.4 Results of Proposed Control Method (Torque to Orientation)

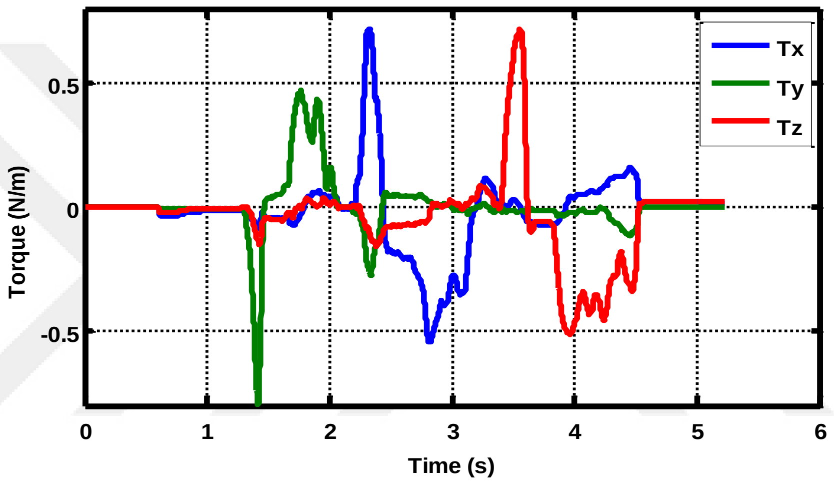
Another experiment is applied to verify the torque to orientation (T/O) control method as illustrated in Figure 6.4. In this experiment, a torque in three directions is applied to the end effector by human hand. First torque is about y-axis followed by torque about x-axis then z-axis of system coordinate. The measured torque by F/T sensor is shown in Figure 6.4(a). The equation (4.19) in Chapter 4 is applied to rotate the torque from F/T sensor coordinate to the base coordinate as shown in Figure 6.4(b). The torque T^0 is converted to the change in orientation $\delta\theta$ by using the equation (4.23). Figure 6.4(c) shows the orientation change.

6.5 Results of Online Motion Learning by HRI

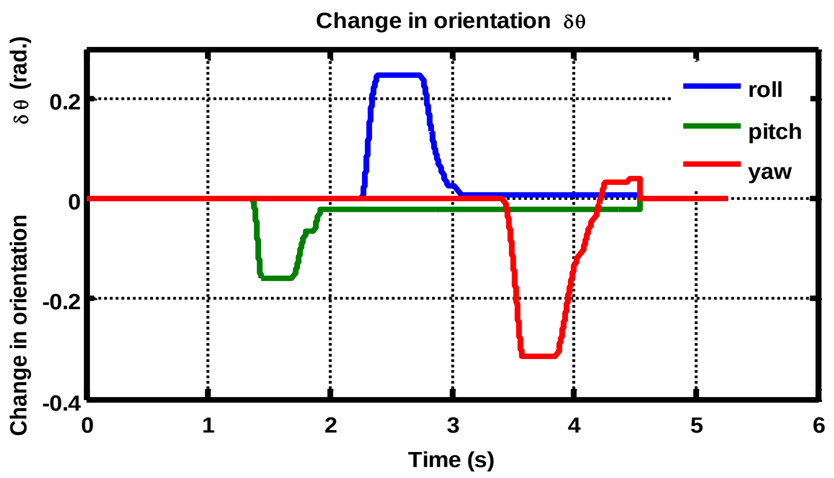
The experiment of online motion learning is used to execute the system performance during completed task. The path motion configuration has covered the most space 3D system workspace. Three different cases are studied. In the first case, the Robotic system is taught by recording joint position. Some differences are seen between the desired path and repeated path. Figure 6.5 shows the desired path and the repeated path in x,y,z axes for Case 1 position recording.



(a)



(b)



(c)

Figure 6.4 T/O verification. (a) Measured torque signals in the sensor's joint, (b) Torque in the frame coordinate, and (c) Change in orientation $\delta\theta$

The case of robotic motion learning by recording joint velocity is shown in Figure 6.6. This case showed significantly better performance. Improvements are seen in the repeated motion. Case 3 is the robotic motion learning by recording F/T signal, the execution of this case gave a perfect matching between the desired path and the repeated path as shown in Figure 6.7.

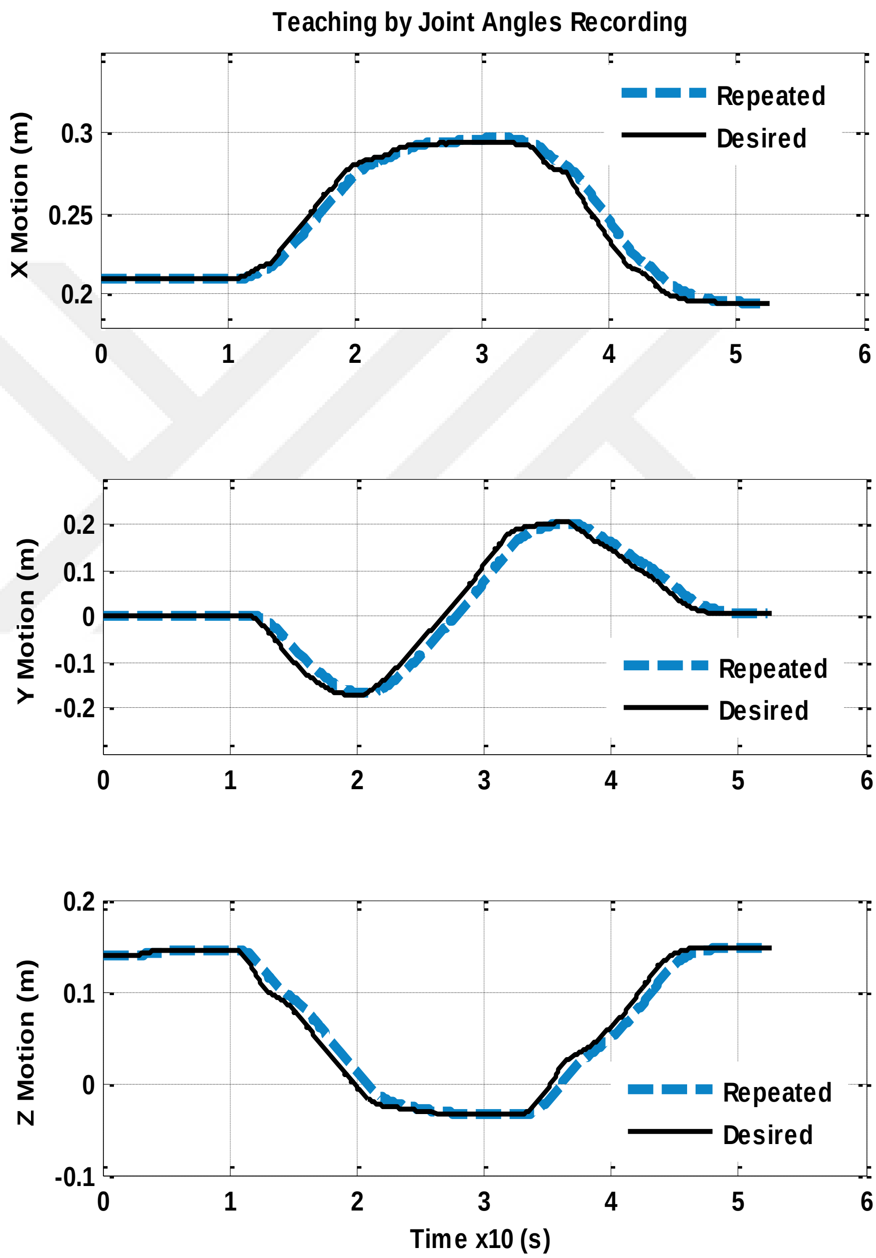


Figure 6.5 Case1 (Robotic motion learning by joint position recording)

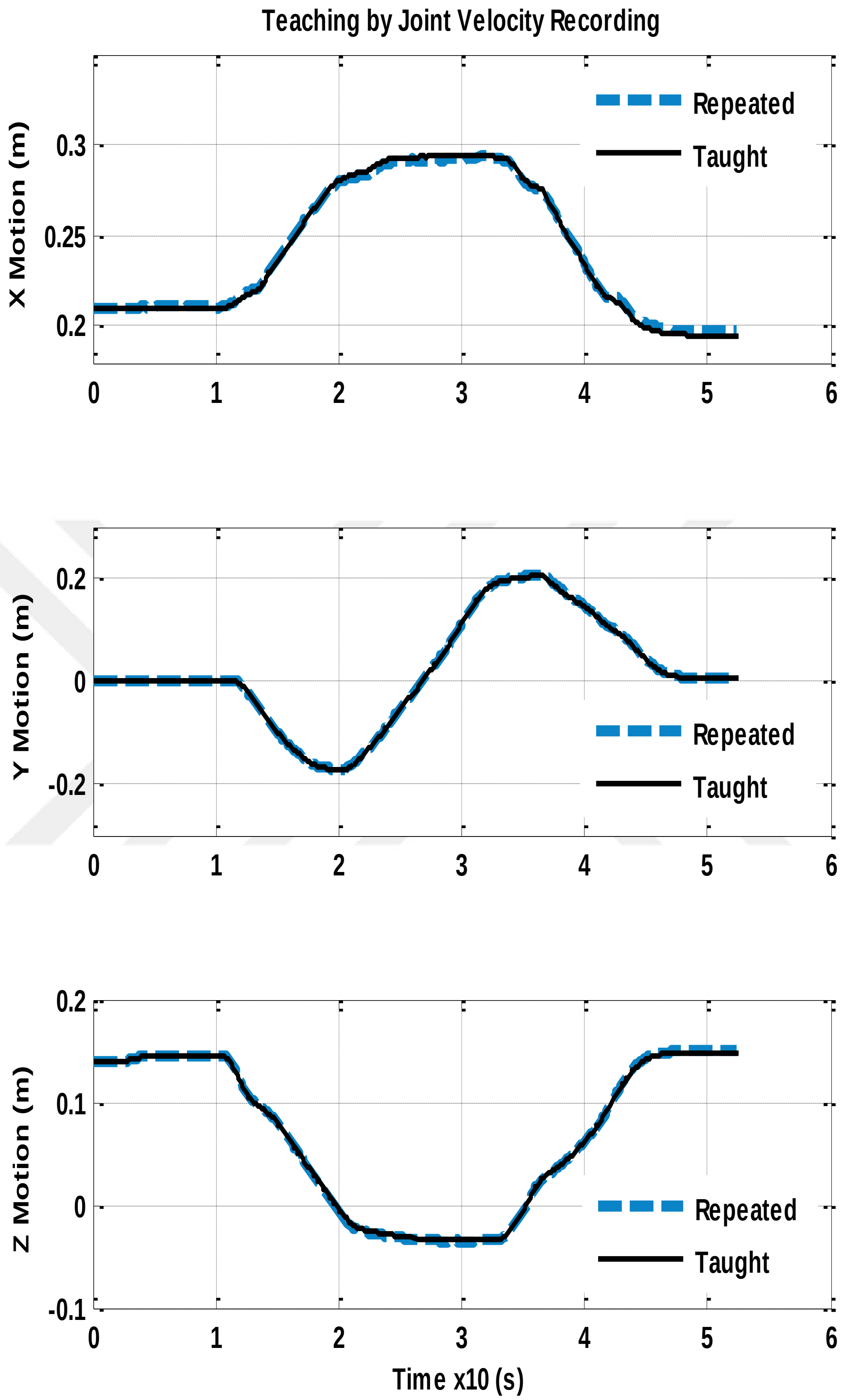


Figure 6.6 Case 2 (Robotic motion learning by joint velocities recording)

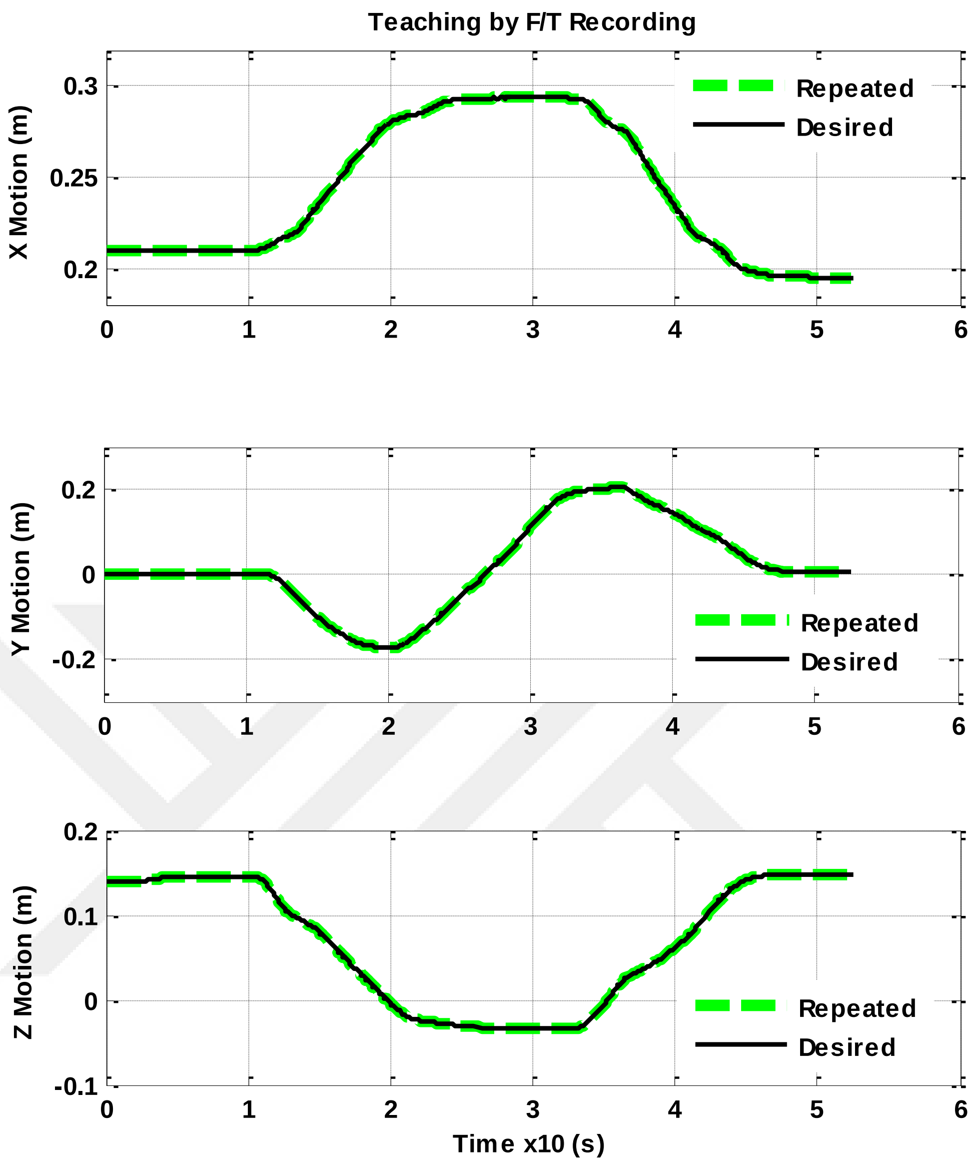


Figure 6.7 Case 3 (Robotic motion learning by F/T signal recording)

The results of online motion learning in repeated path are compared with the desired path. For each case of HRI teaching, the absolute errors (the differences in position) between the desired paths and repeated paths of motion are taken for Cartesian motion. Figure 6.8 presents the absolute error for Case 1; teaching by recording joint position. The maximum error in *x* direction was 1.5cm, and in *y* direction it was 4cm, while in *z* direction it was 2.2 cm. This error is reduced when the joint velocity is recorded as a teaching variable as shown in Figure 6.9. In this case, the maximum error in *x* direction was 1cm, and in *y* direction it was 2.5cm, while in *z* direction it was 1.6 cm. The use of joint positions and joint velocities as teaching variables are applicable in some robotic applications like spray painting robot, but many applications need higher precision such as robotic handling operations. In this study, the F/T signal is the proposed teaching variable because it is the main effective signal during HRI. Referring to Figure 6.10, the absolute error between the desired path and repeated path is illustrated for case of teaching by F/T signal recording. The robot showed a superior performance motion repeating and position error reduction, the maximum error in *x* direction was 0.3 cm, and in *y* direction it was 1cm, while in *z* direction it was 0.8 cm. In this case many robotic arm applications can be implemented by HRI. The robots can be used safely in human life because it has a good response for any interaction force and torque during teaching and task execution.

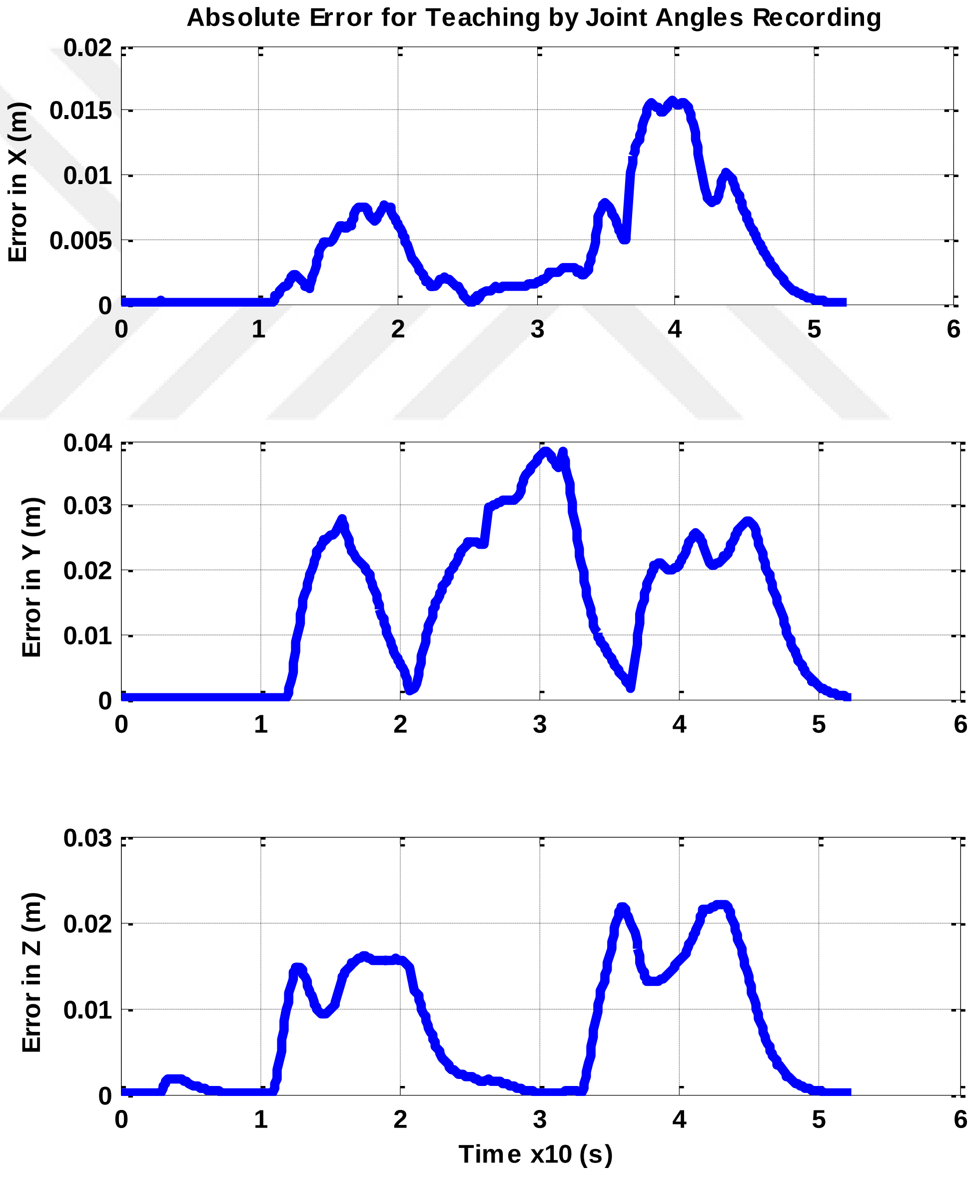


Figure 6.8 Absolute error Case 1 (The teaching by joint position)

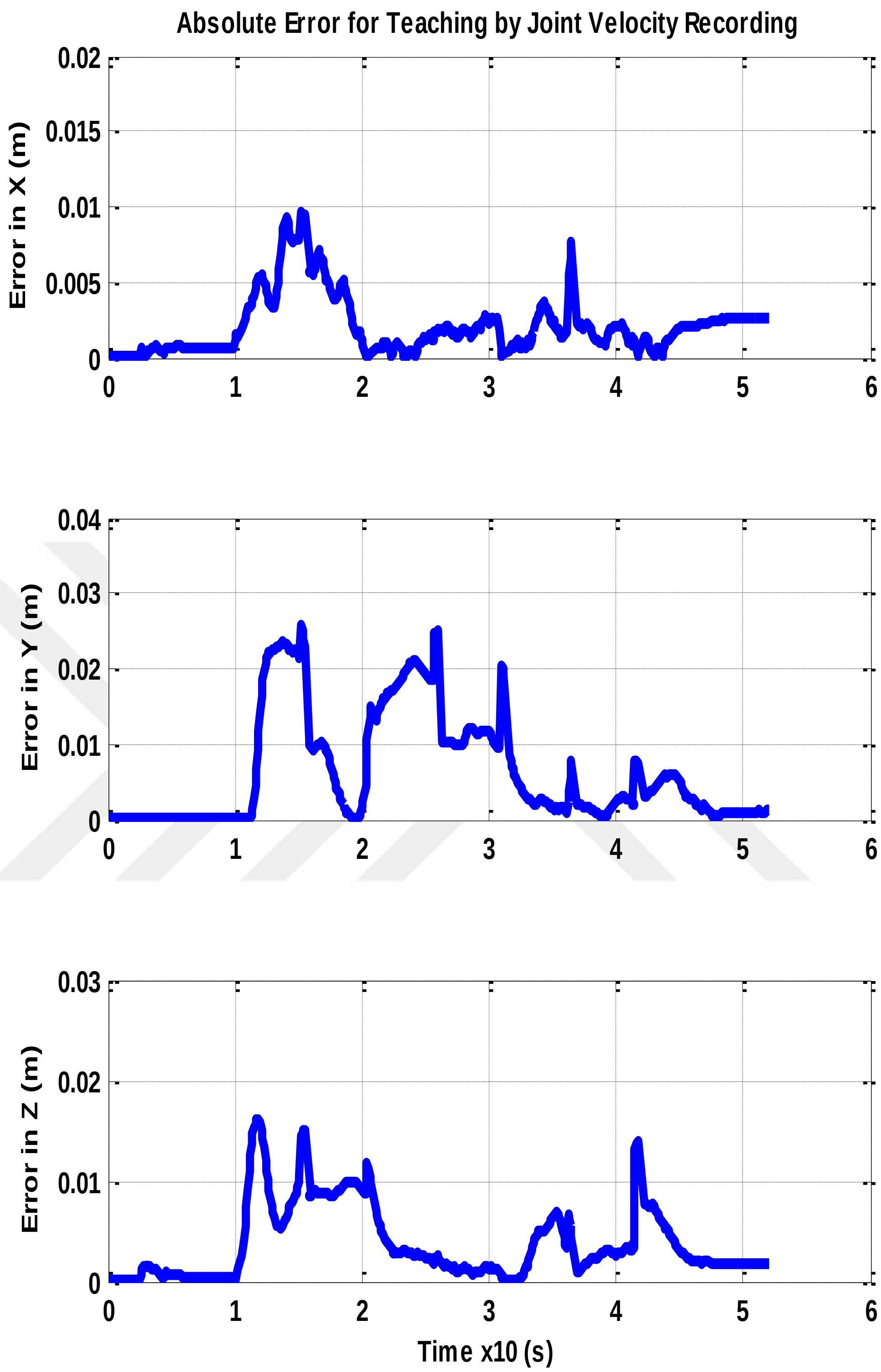


Figure 6.9 Absolute error Case 2 (The teaching by joint velocities)

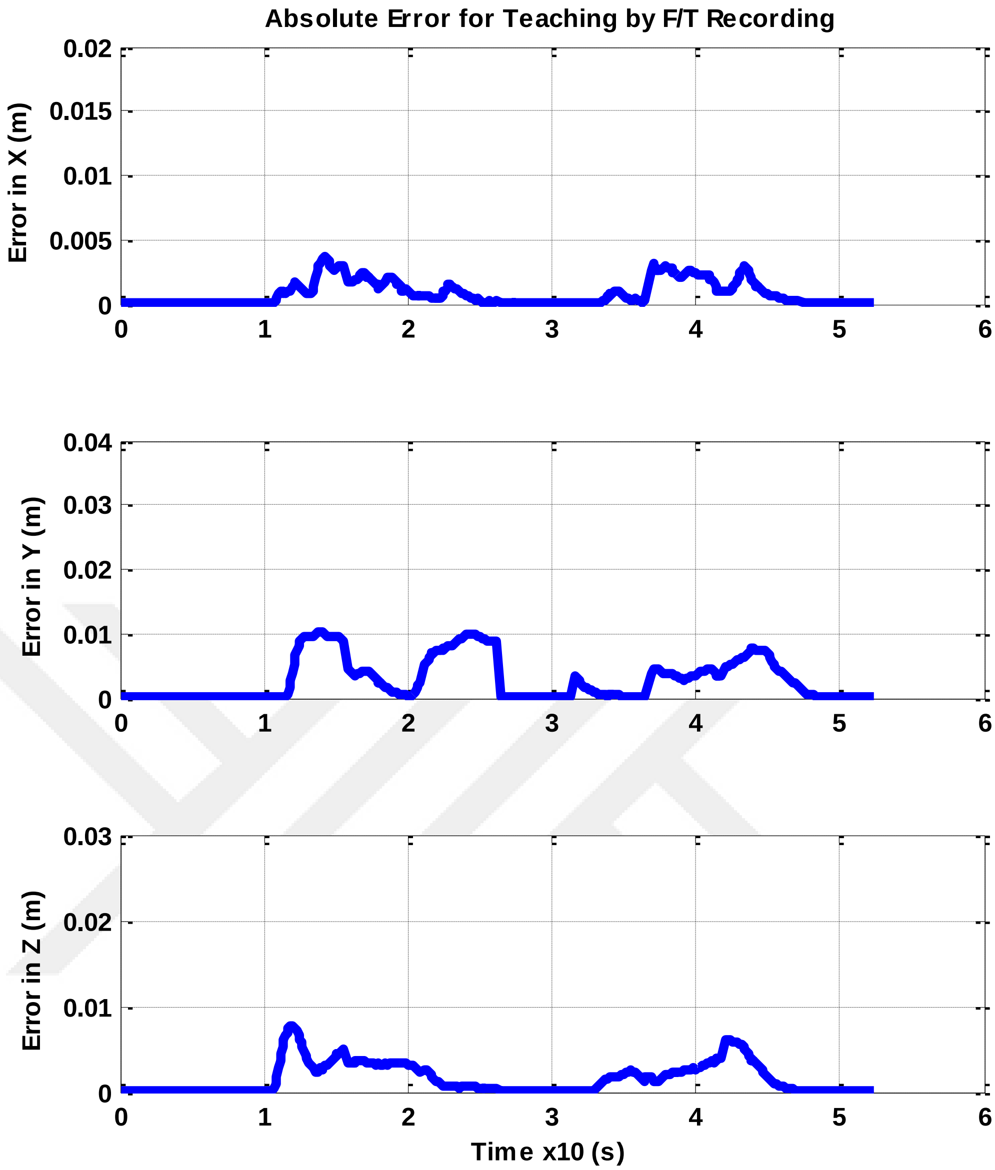


Figure 6.10 Absolute error Case 3 (Proposed teaching by F/T signal)

6.6 Result of ANN for Inverse Kinematic Solution

The performance of ANN is calculated by the differences between the network outputs and target which is known as the mean squared error (MSE). Figure 6.11 shows the performance of the traditional neural network. The MSE of training is decreased until the validation error is stopped at the epoch 121. MSE was 1.1892e-5 in the best performance. A closed relation is between the output and target samples at correlation of 0.99758. The performance of the proposed ANN is shown in Figure 6.12. The MSE of training is decreased rapidly through the learning process, it is decreased until the validation error is stopped at the epoch 68 and MSE is 3.3029e-8 in the best performance. A closed relationship is between output and target samples at correlation 0.99999. The new outputs of the network are checked by the test data, the training samples are 2800 samples, the validated and test samples were 600 for each one. Cartesian space is used in the motion of the robot. In Figure 6.13(a), the motion in x direction is an enlargement of the time period (20s - 25s). Figure 6.13(b) is the motion in y direction enlarged for the time period (26s to 30s). Figure 6.13(c) is the motion in z direction enlarged for the time period (14s to 19s). The differences in x, y, z directions for motion resulted from proposed ANN and traditional ANN. The high accuracy and precision is clearly seen in proposed ANN.

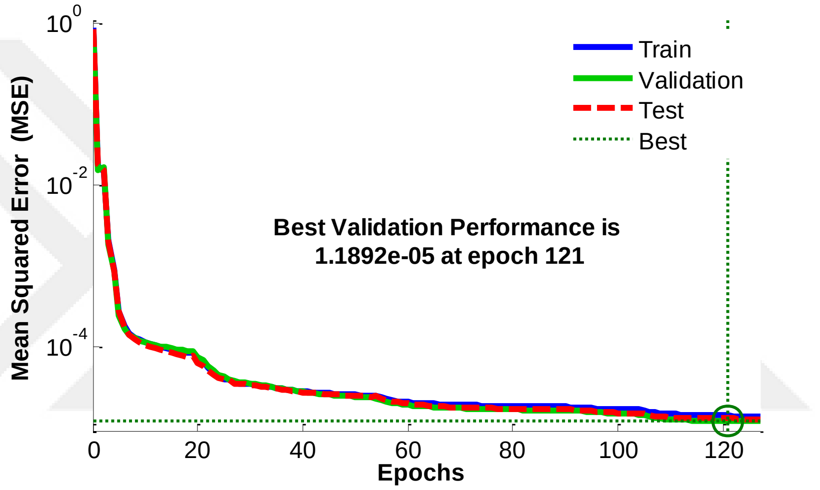


Figure 6.11 Performance of the traditional neural network

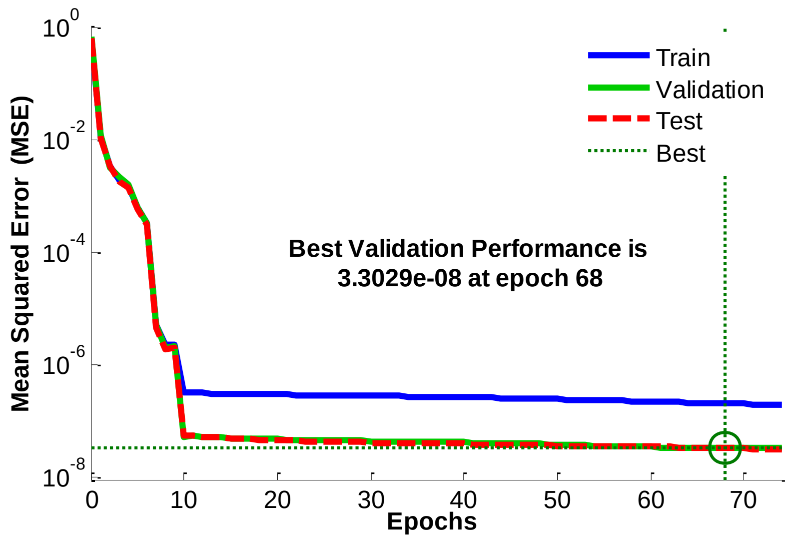


Figure 6.12 Performance of the proposed neural network

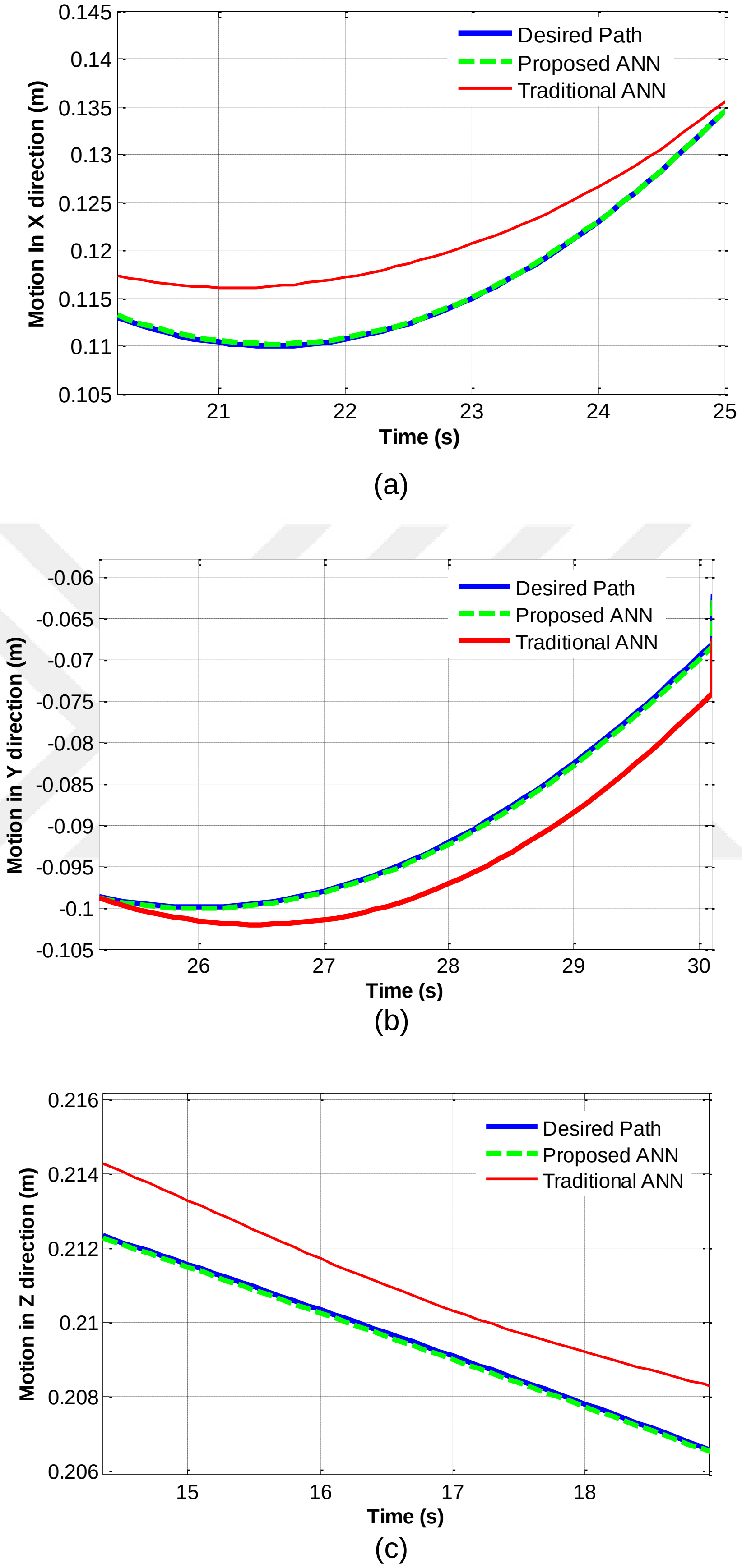


Figure 6.13 Robot motion by two ANNs (a) The motion in x direction, (b) The motion in y direction, (c) The motion in z direction

The differences between desired path and artificial neural network path are measured by calculating the error in position from proposed ANN and traditional ANN with respect to position from desired path. Figure 6.14 shows the error in x, y, z position for proposed ANN, while Figure 6.15 shows the errors in x, y, z position for traditional ANN.

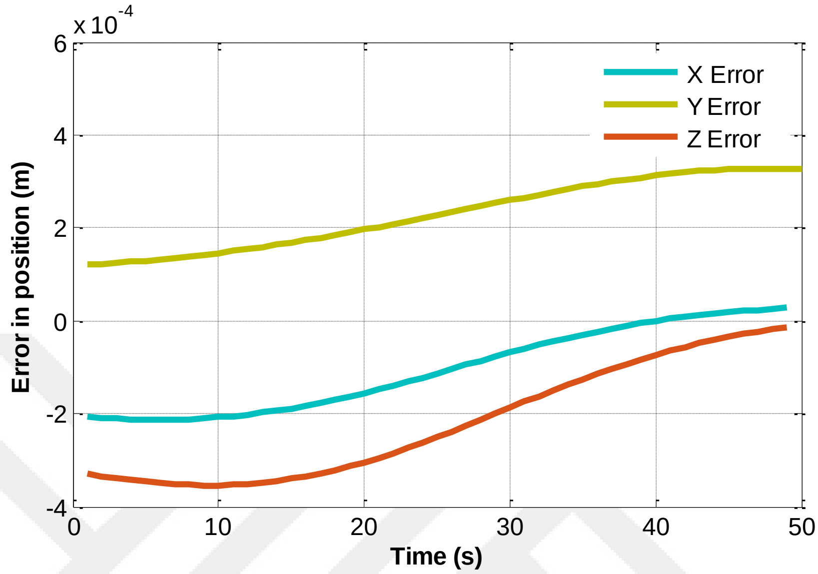


Figure 6.14 Proposed ANN motion error in x, y, z direction

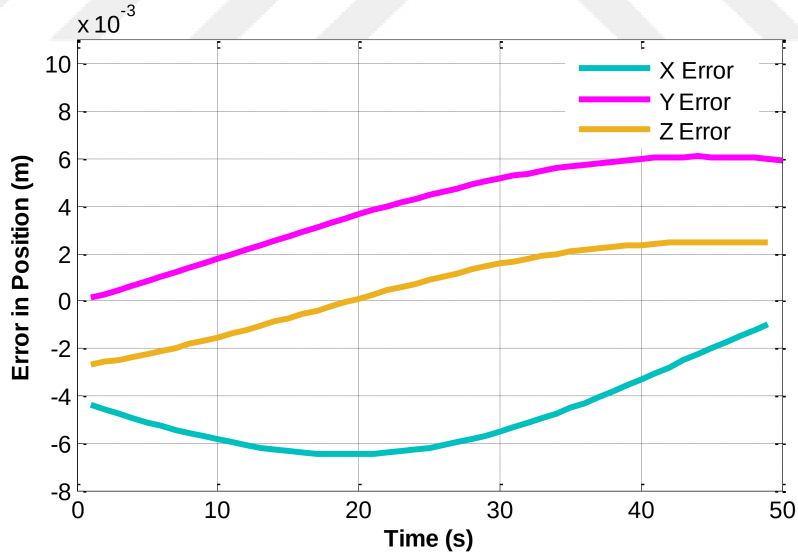


Figure 6.15 Traditional ANN motion error in x, y, z direction

The motion curves of proposed ANN are much more precise than the motion curves of traditional ANN in estimating the joint angles of the robot for desired positions. The maximum error in x direction for proposed design of ANN is near 0.22 mm and for traditional design of ANN2 is 6.5 mm. The maximum error in y direction for proposed design of ANN is less than 0.3 mm and for traditional design of ANN2 is near to 6 mm. The maximum error in z direction for proposed design of ANN is 0.35 mm and for traditional design of ANN2 is near to 2.5 mm. Table 6.3 shows performance of proposed and traditional ANNs in term of the error percentages.

Table 6.3 Performance of robotic system by ANNs

Parameters	***Proposed ANN***	***Traditional ANN***
Px error %	0.17	5.78
Py error %	0.36	7.25
Pz error %	0.12	1.28
MSE	3.3029 e^{-8}	1.1892 e^{-5}
Regression	0.99999	0.99758

The errors are increased in some points and are reduced in other points. High error is found because of estimating position by ANN while the point of low error is the point that is near the samples of training set in ANN. The proposed ANN has given better accuracy and precision in position than the traditional ANN. This method is applicable in precise robotic motion. According to ANN based solving inverse kinematics results in the literature [35-38, 43, 44] the proposed approach in this study has a minimized error in the inverse kinematics solution. Table 6.4 shows system performance compression between this study and other studies from the literature. It can be obviously seen that higher accuracy and lower MSE are obtained in this work.

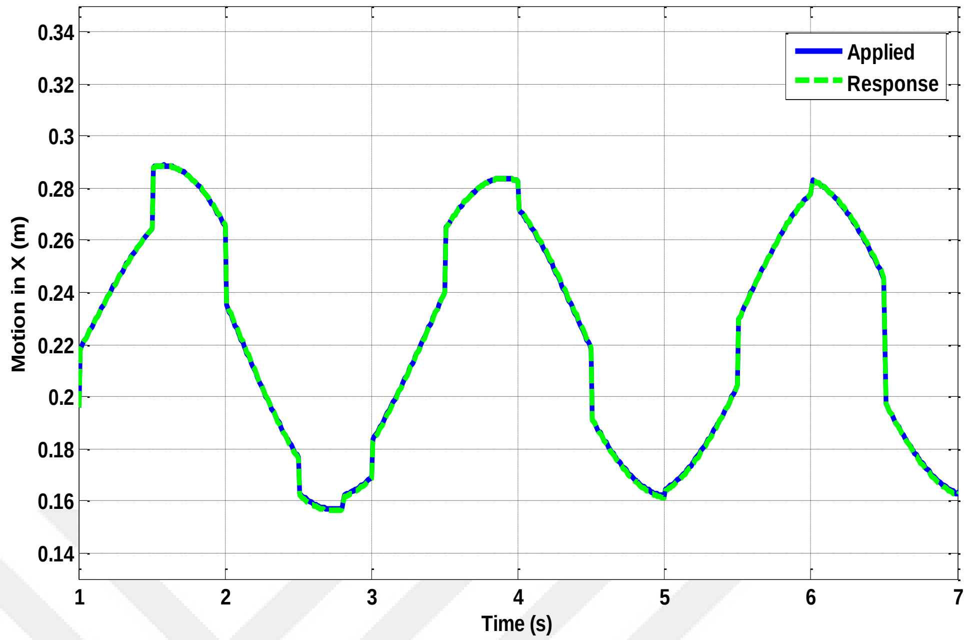
6.7 Results of Robotic Assisted Thoracoscopic Surgery

The motion of Thoracoscope tool is applied as a circular motion through the trocar point. It is achieved by changing the rotational angles (Roll, Pitch) of the tip of Thoracoscope. Since the motion of robot in surgical operation is in Cartesian space. This motion produces a translational path and rotational path. The applied path is compared with the system response. In Figure 6.16(a) and 6.16(b), it is possible to observe the similarity in translational motion for *x, y* directions resulted from proposed control method for robotic surgery in Chapter 4. The differences between desired and response paths are very low. The absolute errors in position form responded and applied motion are calculated to get the system performance during this application. Figure 6.17 shows the error in *x* and *y* positions for experiment of robotic surgery.

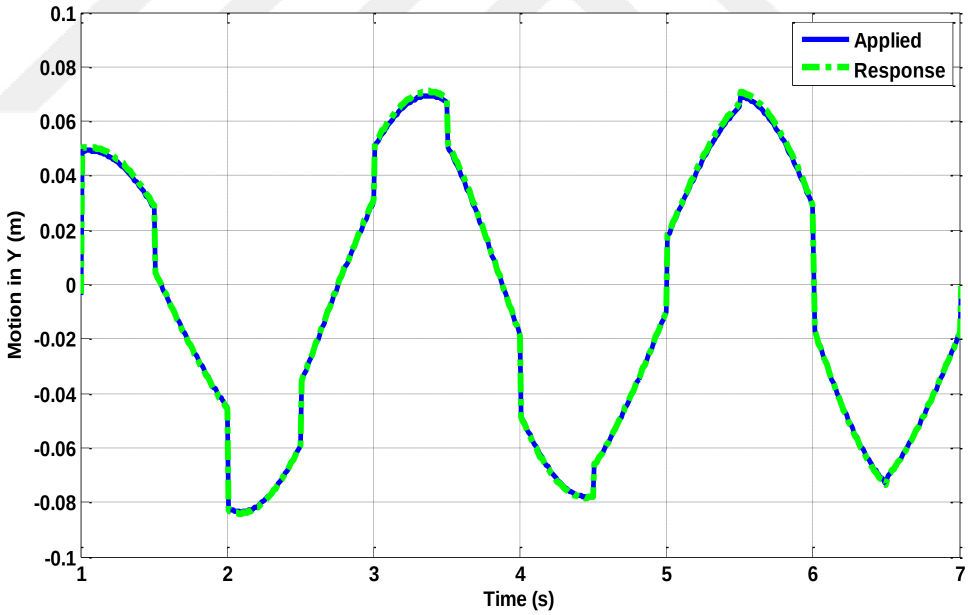
Table 6.4 System performance comparison between this study and other studies mentioned in the literature [81]

Study	System	DOF	Method	MSE	Hidden Layers	Errors
Proposed	Denso	6	ANN	$3.3e^{-8}$	10	x=0.17% y=0.36% z=0.12%
Luv A. et al. 2014 [43]	PUMA 560	6	ANN	1.217	30	x=6.42% y=4.90% z=2.92%
Ali T. et al. 2010 [35]	FANUC M-710i robot	6	ANN	~ 1	1	x=3.34% y=6.72% z=0.35%
Hamid T. et al. 2014 [37]	Simulation PA-10 robot	7	NNs with optimization	1	-	end effector 5 mm
Adrian D. 2014 [44]	Planar simulation	3	NNT	0.0054	1	-
Rasit K. 2013 [37]	Stanford	6	ANN	2.38	25	end effector 4.28 mm
Santosh K. et al.2012 [38]	Simulation	3	ANN, FLANN	>1	20	-
Bassam D. et al. 2010 [38]	Simulation	2	NNT	$5.24e^{-5}$	2	-

The maximum error in x-axis is 8×10^{-4}m. The maximum error in y-axis is 15×10^{-4}m. Another investigation is carried out in this study to realize the behavior of system orientation for Thoracoscope during a surgical operation task. A rotation in x-axis (Roll) and rotation in y-axis (Pitch) are implemented on robots tool (endoscope) by proposed control method. The rotation of Thoracoscopic tool gave a harmonic motion in system response. Figure 6.18(a) and Figure 6.18(a) show the rotational (roll, pitch) motions of the applied path and system responses. The absolute errors in orientation for system with respect to orientation applied path are calculated. Figure 6.19 shows the orientation error in proposed control method for roll and pitch angles. The maximum error in roll orientation is 3.3×10^{-3}rad. The maximum error in pitch orientation is 5.2×10^{-3}rad. The results of experiment proved that the proposed the system capability to implement a surgical manipulation, which method of using imaginary end effector in the control method, has a perfect performance by looking to the results of this experiment, and this is the first study that achieved this behavior for the kinematics of RCM.



(a)



(b)

Figure 6.16 Translational motion of endoscope for robotic surgery
(a) The motion in x direction, (b) The motion in y direction

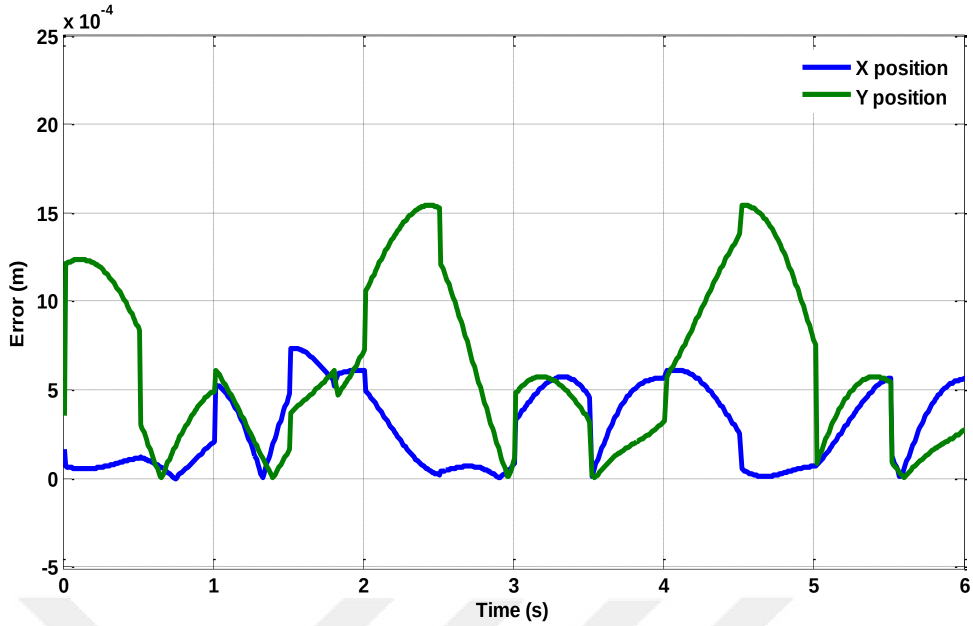
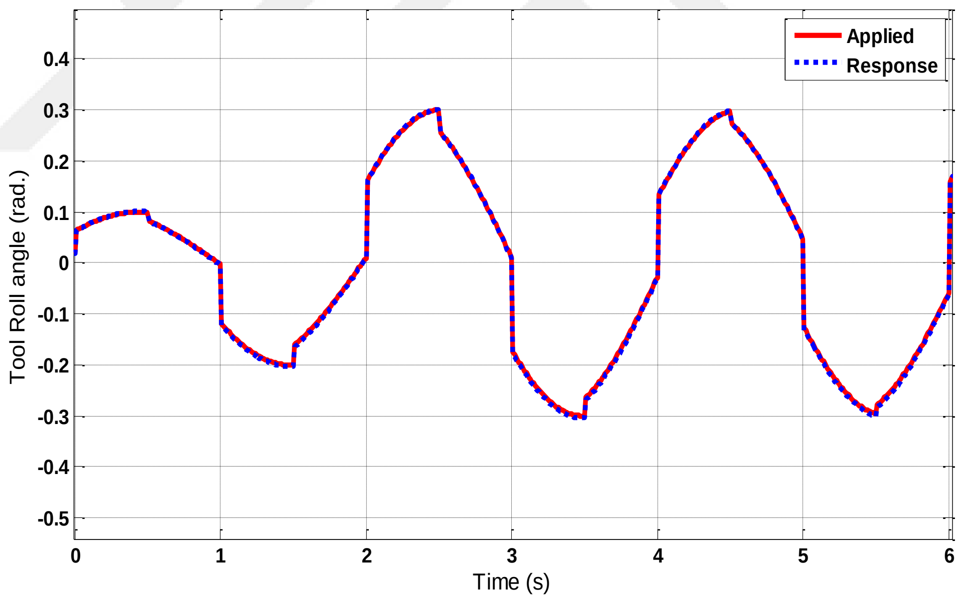
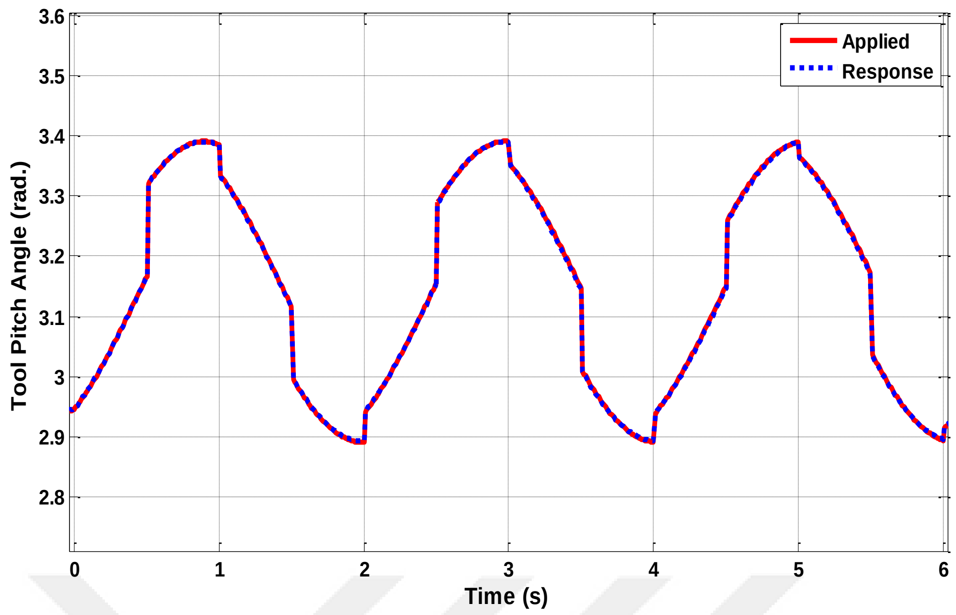


Figure 6.17 Absolute position error in x and y directions



(a)



(b)

Figure 6.18 Rotational motion of Thoracoscope, (a) roll angle rotation in x-axis, (b) pitch angle rotation in y-axis

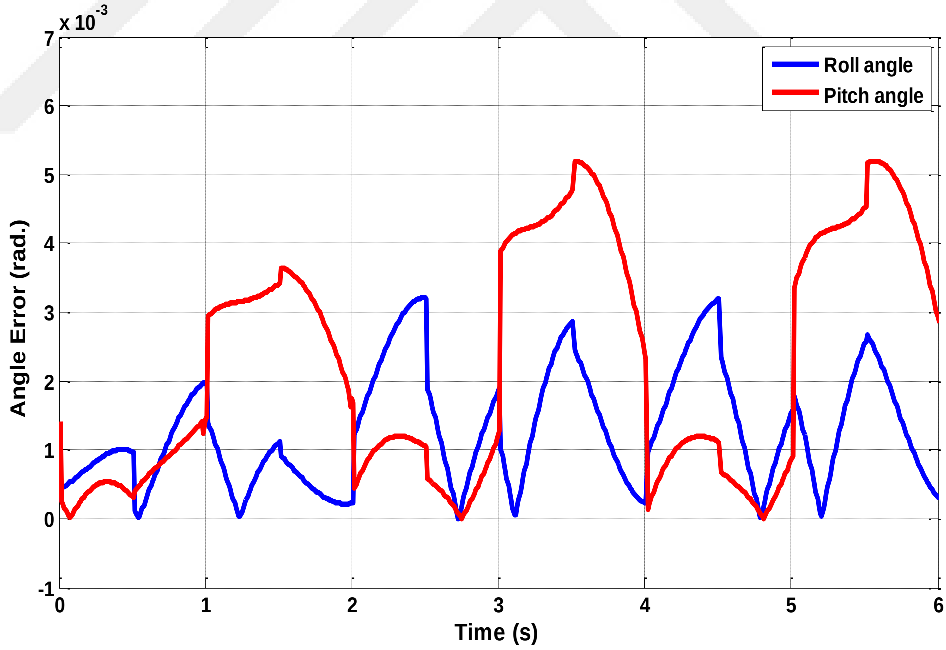


Figure 6.19 Error in rotation about x, y axes (roll, pitch)

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

This chapter presents conclusions of the research described in this thesis. They are introduced in the order of proposed control method, artificial neural network inverse kinematics solution, robotic surgery, and recommendations for future work.

7.1 Conclusions in Robotic Arm Proposed Motion Control

- In this study, the online teaching by HRI was successfully implemented on the robotic system. The control technique of F/T sensing over robotic arm has been applied to eliminate the effect of tool mass and inertia during the online teaching, as the robot moves according to desired motion by touching the robot's end effector.
- The virtual spring control method is modified by the assistance of the designed model in the human-robot interaction implementation. This modified controller is verified by the experimental work. The experimental results of verification revealed that converting the force/torque to motion was very flexible. The effects of gripper inertia during teaching the robotic arm were successfully eliminated.
- It is clear that the F/P control method is not enough to manipulate robot end effector; T/O is strongly required to have full robot manipulation. The position as well as orientation is taught to robot by human hand, the behavior of robot was observed in real time with VRM.
- During online teaching, recording joint angles or joint velocity generated an error in the repeated mode. This error was significantly minimized in this study by recording the effective variables on HRI; external force and torque. Path of taught motion is repeated by robot in execution mode. The repeated path was almost the same as taught path, which was achieved in the experiments mentioned in this study when F/T signal is recorded in teaching mode.
- This is the first study about HRI in which force and torque on end effector were applied in the controller with inertia effect by designed nominal system, while the position error was minimized.

7.2 Conclusions in ANN Inverse Kinematic Solution

- This study introduces a very accurate solution for inverse kinematics by using ANN to overcome the drawbacks of traditional ANN controller. A new design of ANN has been proposed for the optimal robot motion control in Cartesian coordinates. In order to evaluate the integral performance of the system, the current joint angles information was added to the traditional ANN based on inverse kinematics solution. The proposed design showed improvement in performance of end effector in some aspects.
- The motion of robot has been executed, which satisfied the constraints of robotic arm motion by the designed structure. Robot kinematics was analyzed, and P/O of end effector in different configurations was studied.
- A comparison has been carried out between two ANNs. The parameters of motion and the errors were calculated. The results showed that the proposed ANN has superior performance in terms of estimation of joint angles. The design of ANN compared to other techniques is applicable for some of the most difficult and challenging problems of kinematics. These results have proved the effectiveness of the proposed ANN. The inclusion of current configuration of joint angles in ANN increased the accuracy of ANN estimation, and succeeded in mapping between input position and joint angles output.

7.3 Conclusions in Robotic Surgery

In order to overcome the drawbacks of traditional RAMIS controller, the following conclusions are obtained:

- A new remote center of motion kinematics is proposed. Motion control method has been introduced for the kinematics of RCM mechanism. In order to evaluate the integral performance of the system, it was evaluated by experimental test. The proposed design showed improvement in performance of endoscope in some aspects.
- The motion of robot in RATS has been presented, which satisfied the incision motion and constraints of RCM by the designed structure. Kinematics were analyzed, and position/orientation (P/O) of endoscope in different configurations were studied.

- Inclusion F/T feedback for motion control prevented the body skin from tool pressure during manipulation in the robotic assisted Thoracoscopic surgery.
- A model of the minimally invasive surgery robot has been set up based on the requirements of RCM. The real robot performances were presented in this study.

7.4 Recommendations for Future Work

For future work, the following recommendations are introduced in this study:

- The online teaching by HRI can be implemented on virtual robotic system. The control technique of F/T sensing can be studied to sense forces for virtual reality objects.
- The virtual spring control method can be modified by using virtual spring damper design to control the human-robot interaction implementation. The control methods of F/P and T/O are applied. Different control methods of force to velocity (F/V) and torque to angular velocity (T/W) can be studied. The robot response will be measured and can be compared with present control method.
- For online teaching, recording joint angle or joint velocity generates an error in the repeated mode. An adaptive technique can be used to combine external force with recorded joint variable.
- The robotic system can be developed with robotic vision system and image processing analysis for system motion control.
- A solution for Jacobian matrix can be introduced by using the artificial neural network to overcome the drawbacks of traditional ANN controller. Jacobian manipulation design can be used for RCM in robotic surgery.

REFERENCES

[1] Dominic L., Gianluca G., Florian P., Christian O., and Alin A. (2013). Modal limit cycle control for variable stiffness actuated robots, *IEEE International Conference on Robotics and Automation*, *IEEE*, 4934-4941.

[2] Clint H. (2010). Human robot interaction and future industrial robotics applications, I*EEE/RSJ International Conference on Intelligent Robots and Systems, IEEE*, 4749-4754.

[3] Hagenah H., Böhm W., Breitsprecher T., Merklein M., and Wartzack S. (2013). Modelling, construction and manufacture of a lightweight robot arm, *ScienceDirect, Procedia CIRP*, **12**, 211-216.

[4] Anisur R., Alimul H. K., Tofayel A., and Mohsin S. (2013). Design, analysis and implementation of a robotic arm - animator, *American Journal of Engineering Research (AJER)*, **2**(10), 298-307.

[5] Varalakshmi B. D., Thriveni J., Venugopal K. R., and Patnaik L. M. (2012). Haptics: state of the art survey, *IJCSI International Journal of Computer Science*, **9**(5), 234-244.

[6] Jainish S. Kothari, and Tanay S. Vaidya. (2014). Analog haptic robotic arm, *International Journal of Scientific & Engineering Research*, **5**(1), 771-774.

[7] Arnab Das, Swagat Kumar S., and Anil Kumar P. (2012). Operative precept of robotic arm expending haptic virtual system, *IOSR Journal of Engineering (IOSRJEN)*, **2**(11), 37-43.

[8] Reshamwala1 A., and Singh R. (2015). A review on robot arm using haptic technology, *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, **4**(4), 2204-2211.

[9] Pholchai C., Dalong W., Ngaiming K., and Dikai L. (2008). A haptic base human robot interaction approach for robotic grit blasting, The *25th International Symposium on Automation and Robotic in Construction*, 148-154.

[10] Mehmet T. (2009). A human robot interaction application for robotic education, *Sciencedirect, Procedia Social and Behavioral Sciences*, **1**, 2164–2169.

[11] Mustafa Suphi E., and Aude B. (2015). End-point impedance measurements across dominant and non-dominant hands and robotic assistance with directional damping, *IEEE Transactions on Cybernetics*, **45**(6), 1146-1157.

[12] Michael A. Goodrich, and Alan C. Schultz. (2007). Human–robot interaction: a survey, *Foundations and Trends in Human–Computer Interaction*, **1**(3), 203–275.

[13] Bassam Daya, Shadi Khawandi, and Mohamed Akoum. (2010). Applying neural network architecture for inverse kinematics problem in robotics, *J. Software Engineering & Applications*, **3**, 230-239.

[14] Ali T. Hasan, and H. M. Al-Assadi. (2010). Performance prediction network for serial manipulators inverse kinematics solution passing through singular configurations, *International Journal of Advanced Robotic Systems*, **7**(4), 11-24.

[15] Pedro Q., Rui C., and Cristóvão S. (2010). Haptic tele-manipulation for robotic assisted minimally invasive surgery with explicit posture control, *IEEE 18th Mediterranean Conference on Control & Automation*, 808-814.

[16] Staub C., Can S., Jensen B., Knoll A., and Kohlbecher S. (2012). Human-computer interfaces for interaction with surgical tools in robotic surgery, The Fourth *IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*, 81-86.

[17] Valentina V., Su-Lin L., Thomas P. Cundy, and Guang-Zhong Y. (2013). Emerging Robotic Platforms for Minimally Invasive Surgery, *IEEE Reviews in Biomedical Engineering*, **6**, 111-126.

[18] Seung Yeol L., Kye Young L., Sang Heon L., JinWoo K., and Chang Soo H. (2007). Human-robot cooperation control for installing heavy construction materials, *Autonomous Robots*, **22**(3), 305-315.

[19] Ioannis I. (2013). Utilizing artificial skin for direct physical interaction," *in Proc. IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 810-815.

[20] Seung Hoon L., Cheng Jie L., Dong Hyung K., Jin Ho K., and Chang Soo H. (2009). The direct teaching and playback method for robotic deburring system using the adaptive force- control, *In Proc. IEEE, International Symposium on Assembly and Manufacturing*, 235-241.

[21] Daisuke K., Masatoshi N., Satoru G., and Nobuhiro K. (2001). Human direct teaching of industrial articulated robot arms based on force-free control, *Artificial Life and Robotics*, **5**(1), 26-32.

[22] Hooman L., and Joongbae K. (2013). A survey on robotic teaching: categorization and brief review, *Applied Mechanics and Material*, **330**, 648-656.

[23] Dana K, and Elizabeth C. (2007). Pre-collision safety strategies for human-robot interaction, *Autonomous Robots*, **22**, (2), 149-164.

[24] Ana Lucia Pais U., Keisuke U., Yoshihiko N., and Aude B. (2015). Task parameterization using continuous constraints extracted from human demonstrations, *IEEE Transactions on Robotics,* **31**(6), 1458-1471.

[25] Simon P., Ana Luisa T., and Rajni V. Patel. (2010). A 7-DOF haptics-enabled teleoperated robotic system: kinematic modeling and experimental verification, *In Proc. 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics,* 906–911.

[26] Dong Hyuk L., Uikyum K., and Hyouk Ryeol C. (2014). Development of multi-axial force sensing system for haptic feedback enabled minimally invasive robotic surgery, *In Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 4309–4314.

[27] Yiannis K., Dimitrios P., and Zoe D. (2016). A model-free controller for guaranteed prescribed performance tracking of both robot joint positions and velocities, *IEEE Robotics and Automation Letters*, **1**(1), 267-273.

[28] Leonidas D., and Zoe D. (2015). Haptic rolling contact motion generation and control of robotic fingers, *Journal of Intelligent & Robotic Systems*, 1-18.

[29] Jin Tak K., Jongwon L., Hyogon K., Jaehong S., Sang-uk C., Byungyun P., Sangdeok P., and Jungsan C. (2015). Method for improving the position precision of a hydraulic robot arm: dual virtual spring–damper controller, *Intelligent Service Robotics*, 1-7.

[30] Jilin Z., Franc O. M., Emil M. P., and Nicolas D. Georganas. (2009). Compensating device inertia for 6-dof haptic rendering, *in Proc. IEEE International Conference on Systems*, 12-17.

[31] Takahiro E., Mana K. and Haruhisa K. (2013). A finger skill transfer system using a multi-fingered haptic interface robot and a hand motion image, *Robotica journal*, **31**(8), 1251-1261.

[32] Hwang G., Preeda C., Hashimoto H. (2007). Developing intelligent robotic biomanipulation system using haptic interface, *In proc. 2nd IEEE International Conference on Nano/Micro Engineered and Molecular Systems*, 396–401.

[33] Paul E. R., Jeremy S., Kevin Y., and Manuela V. (2008) Using dialog and human observations to dictate tasks to a learning robot assistant, *Intelligent Service Robotics*, **1**(2), 159-167.

[34] Takehiko O., and Hajime K. (2010). Solution for ill-posed inverse kinematics of robot arm by network inversion, *Hindawi. Journal of Robotics*, 1-9.

[35] Ali H., Ismail N., Hamouda A., Aris I., Marhaban M., and Al-Assadi H. (2010). Artificial neural network-based kinematics Jacobian solution for serial manipulator passing through singular configurations, *Advances in Engineering Software*, **41**, 359–367.

[36] Rasit K. (2013). A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization, *Information Sciences*, **222**, 528–543.

[37] Hamid T., and Mohammad F. (2014). Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: A Lyapunov-based approach, *Robotics and Autonomous Systems*, **62**, 766–78.

[38] Santosh N., Swetalina P., Raj Sekhar S., and Ranjan D. (2012). A novel application of artificial neural network for the solution of inverse kinematics controls of robotic manipulators, *I. J. Intelligent Systems and Applications*, **9**, 81-91.

[39] Bassam D., Shadi K., and Mohamed A. (2010). Applying neural network architecture for inverse kinematics problem in robotics, *J. Software Engineering & Applications*, **3**, 230-239.

[40] Aixuan W., Zhiping S., Yongdong L., Minhua W., Yong G., Jie Z., and Hongxing W. (2016). Formal kinematic analysis of a general 6R manipulator using the screw theory, *Hindawi, Mathematical Problems in Engineering*, doi.10.1155/2015/549797.

[41] Zongxing L., Chunguang X., Qinxue P., Xinyu Z., and Xinliang L. (2015). Inverse kinematic analysis and evaluation of a robot for nondestructive testing application, *Hindawi, Journal of Robotics*, doi.10.1155/2015/596327.

[42] Guanglei W. (2014). Kinematics and dynamics of an asymmetrical parallel robotic wrist, *Hindawi, Journal of Robotics*, doi.10.1155/2014/296250.

[43] Luv A., Kush A., and Ruth J. (2014). Use of artificial neural networks for the development of an inverse kinematic solution and visual identification of singularity zone(s), *Procedia CIRP*, 812–817, doi.10.1016/j.procir.2014.01.107.

[44] Adrian V. D. (2013). Neural network based inverse kinematics solution for trajectory tracking of a robotic arm, *Procedia Technology*, **12**, 20–27, doi:10. 1016/j.protcy.2013.12.451.

[45] Alejandro S., Ernesto R., Rogelio S., Gordillo J. L., and Leonardo G. (2015). Decoupled closed-form solution for humanoid lower limb kinematics, *Hindawi, Mathematical Problems in Engineering*, doi.10.1155/2015/437979.

[46] David A. E., Rene d. J., Arturo Y., Luis M., and Roque A. (2016). Vibration suppression for improving the estimation of kinematic parameters on industrial robots, *Hindawi, Shock and Vibration*, doi.10.1155/2016/6954012.

[47] Peidong L., Lianzheng G., Yihuan L., Lijun Z., Ruifeng L., and Ke W. (2016). An augmented discrete-time approach for human-robot collaboration, *Hindawi Discrete Dynamics in Nature and Society*, doi.10.1155/2016/9126056.

[48] Kurt E. C., and Ying S. (2010). A geometric approach for robotic armkinematics with hardware design, electrical design, and implementation, *Hindawi, Journal of Robotics*, doi.10.1155/2010/984823.

[49] Guanglong D., and Ping Z. (2013). IMU-based online kinematic calibration of robot manipulator, *Hindawi, The ScientificWorld Journal*, doi.org/10.1155/2013/139738.

[50] Daniel P., Konrad K., and Grzegorz K. (2014). Kinematics analysis of a novel five-degree-of-freedom spatial parallel micromanipulator, *Hindawi, Journal of Robotics*, doi.10.1155/2014/806294.

[51] Kishore K., Srinath A., Jugalanvesh G., Premsai R., and S M.. (2013). Kinematic analysis and simulation of 6dof kukakr5 robot for welding application, *International Journal of Engineering Research and Applications*, **3**(2), 820-827.

[52] Mahidzal D., and Jian-Ding T. (2011). Forward and inverse kinematics model for robotic welding process using kr-16ks Kuka robot, *In Proc. 4th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO) IEEE*, doi.10.1109/ICMSAO.2011.5775598.

[53] Ki-Young K., Ho-Seok S., Jung-Wook S., and Jung-Ju L. (2013) A novel surgical manipulator with workspace-conversion ability for telesurgery, *IEEE/ASME Trans Mechatronics*, **18**(1), 200-211.

[54] Jianmin L., Shuxin W., Xiaofei W., and Chao H. (2010). Optimization of a novel mechanism for a minimally invasive surgery robot, *International Journal of Medical Robotics and Computer Assisted Surgery*, **6**, 83-90.

[55] Zhang Z., Yu H., and Du Z. (2015). Design and kinematic analysis of a parallel robot with remote center of motion for minimally invasive surgery. *In Proceedings of the IEEE Int. Conf. on Mechatronics and Automation (ICMA)*, 698-703.

[56] Mohsen M. D., and Bijan S. (2012). Remote centre-of-motion control algorithms of 6- rcrr parallel robot assisted surgery system (PRAMiSS), *In Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 3401-3406.

[57] Youngjin M., Hyuk Jae C., Joon Beom S., and Jaesoon C. (2014). Design and kinematic analysis of a new end-effector for a robotic needle insertion-type intervention system, *International Journal of Advanced Robotic Systems*, **14**, 1-10.

[58] Nastaran A., Milad G., Navid S., Marilena V., and Giuseppe O. (2013). Task control with remote center of motion constraint for minimally invasive robotic surgery, *In Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 5807-5812.

[59] Yili F., Guojun N., Bo P., Kun L., and Shuguo W. (2013). Design and optimization of remote center motion mechanism of minimally invasive surgical robotics, *In Proceedings of IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 774-779.

[60] Roderick C., Locke O., and Rajni P. (2007). Optimal remote center-of-motion location for robotics-assisted minimally-invasive surgery, *In Proceedings of IEEE International Conference on Robotics and Automation*, 1900-1905.

[61] Hyundo C., Ho-Seong K., Yo-An L., and Hyung-Joo K. (2014). surgical robot for single-incision laparoscopic surgery, *IEEE Trans Biomedical Engineering*, **61**(9), 2458-2466.

[62] Cong D. P., Fernando C., Antonio C. L., Fernando L., Pal J. F., and Rolf J. (2015). Analysis of a moving remote center of motion for robotics-assisted minimally invasive surgery, *In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1440-1446.

[63] Doina P., Andras S.,Calin V., and Nicolae P. (2013). Kinematics and workspace modeling of a new hybrid robot used in minimally invasive surgery, *Robotics and Computer-Integrated Manufacturing*, **29**, 463-474.

[64] Yang L., Chng C., Chui C., and Lau D. (2010). Model-based design analysis for programmable remote center of motion in minimally invasive surgery, *In*

Proceedings of IEEE Conference on Robotics, Automation and Mechatronics, 84-89.

[65] Beira R., Santos-Carreras L., Rognini G., Bleuler H., and Clavel R. (2011). A novel remote-center-of-motion parallel manipulator for minimally invasive surgery, *Applied Bionics and Biomechanics*, **8**, 191-208.

[66] Quanser. (2014). industrial open-architecture robotic platforms. Available at: http://www.quanser.com/Products/robotics. Accessed 18 August 2014.

[67] ATI Industrial. (2014). Force/Torque Sensors. Available at: http:// www.ati-ia.com/products/ft/ft_ModelListing.aspx. Accessed 18 July 2014.

[68] Hong D. H., Kim Y. G., Kwak J. H., Jeong C. P., and An J. (2012). Miniature force-torque sensor using semiconductor strain gauge sensor frame design and analysis for development, *Ubiquitous Robots and Ambient Intelligence (URAI), 2012 9th International Conference*, 627-628. doi: 10.1109/URAI.2012.6463101.

[69] Beyeler F., Muntwyler S., Nagy Z., Moser M., and Nelson B. J. (2007). A multi-axis MEMS force-torque sensor for measuring the load on a microrobot actuated by magnetic fields, *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3803-3808. doi: 10.1109/IROS.2007.4399004.

[70] Loske J., and Biesenbach R.. (2014). Force-torque sensor integration in industrial robot control, *Research and Education in Mechatronics (REM) 15th International Workshop*, 1-5. doi: 10.1109/REM.2014.6920241.

[71] Schunk. (2014). Gripping systems parallel gripper. Available at: http://de.schunk.com/de_en/gripping-systems/category/gripping-systems/schunk-grippers/parallel-gripper. Accessed 20 September 2014.

[72] Mehmood N., Ijaz F., Murtaza Z., Ali Shah S.I. (2014). Analysis of end-effector position and orientation for 2P-3R planer pneumatic robotic arm, *in iCREATE 2014 Int. Conf.*, 47-50.

[73] John J. C. (2005). Introduction to Robotics, Mechanics and Control. 3: Pearson Education International.

[74] Corke P. (2011). Robotics, Vision and Control Fundamental Algorithms in MATLAB. 73: Springer-Verlag Berlin Heidelberg.

[75] Park J. H., Oh Y., Kim D., Choi Y., and Yang W. J. H. (2015). Task space control considering passive muscle stiffness for redundant robotic arms, *Intelligent Service Robotics*, **8**(2) 93-104.

[76] Arimoto S., and Sekimoto M. (2006). Experimental Study on Reaching Movements of Robot Arms with Redundant DOFs Based upon Virtual Spring-Damper Hypothesis," i*n Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 562-567.

[77] Valentina V., Su-Lin L., Thomas P. C., and Guang-Zhong Y. (2013). *emerging robotic platforms for minimally invasive surgery, IEEE reviews in biomedical engineering*, 1-6.

[78] Jeffrey A. M. (2003). Advanced Thoracoscopic procedures are facilitated by computer-aided robotic technology, *European Journal of Cardio-thoracic Surgery*, 883-887.

[79] Ovunc B. (2015). Advanced Techniques in Minimally Invasive and Robotic Colorectal Surgery, *Springer Science & Business Media*, DOI 10.1007/978-1-4899-7531-7.

[80] Thomas S. (2004). General Thoracic Surgery. Lippincott Williams and Wilkins.

[81] Almusawi A. R. J, Dülger L. C., and Kapucu S. (2016). A New Artificial Neural Network Approach in Solving Inverse Kinematics of Robotic Arm (Denso VP6242), *Computational Intelligence and Neuroscience*, **2016**, 1-10, doi:10.1155/2016/5720163.

APPENDIX A

DENSO ROBOT SPECIFICATION

A.1 Introduction

The Denso 6-Axis robot shown in Figure A.1 consists of a Denso VP6242G and a Quanser open-architecture control module which has all the capabilities of an industrial system and is interfaced to QUARC. This module contains 6 amplifiers and built-in FF (feed-forward) and PID (proportional, integral, derivative) controllers. The controllers are operating about each motor at a rate of 1kHz. All the gains in the built-in controller are accessible from QUARC blockset. These blocks also have direct access to the amplifiers current commands in the module. The user can both tune the built-in controller gains in QUARC interface or she/he can design their own controller in Simulink environment and command the amplifier currents directly in a fully open-architecture scenario.

Figure A.1 Open-architecture 6-axis robot enabled by Quanser

In the fully open-architecture mode, user-defined currents are sent directly to the motors and the operator is responsible for implementing a stable feedback system. CAUTION: Extra care must be taken when operating the robot in open archetecture mode. The designed or modified controller should always include safety precautions to avoid damage to the robot and the personnel if the controller goes unstable. The general system description, component nomenclature, specifications, model parameters are given in Section A.2, Section A.3 goes through several laboratory requirements and lists various documents that are referenced in this section. Wiring of different components and quick setup is also given. An overview of various supplied files with the robot is given of the Open-Architecture Setup Guide. The procedure to run the supplied open-architecture controllers on the actual robot system is discussed in the same section as well.

A.2 Denso Robot Components

To install Denso robot in open-architecture, the following hardware and software are required:

1. Multi-Axis ATI Force/Torque Sensor
2. QUARC User Manual (type doc QUARC in Matlab to access).
3. QUARC Installation Guide.
4. Denso Robot General Information.
5. Denso Robot Safety Precautions.
6. Denso Robot Installation and Maintenance Guide.
7. National Instruments 16bit Data Acquisition Board.
8. Serial Port Card Installation Guide.

Table A.1 Main Components of the Denso system

Component	Description
Denso 6-axis robot Denso	Robotic arm with compact control module as shown in Figure A.1.
PC with QUARC Software	PC with the QUARC-Simulink configuration.
Force-Torque Sensor	ATI multi-axis force/torque sensor system as described.

A.3 Denso 6-Axis robot

The Denso 6-Axis robot system includes the 6 DOF robotic arm and the control module. This document also assumes that the robot includes the optional force-torque sensor.

Table A.2 Robot specifications of the VP-G series [66].

Item		Specifications	
		6-axis type	5-axis type **(Note 1)**
Model name of robot set **(Note 2)**		VP-6242G	VP-5243G
Model name of robot unit		VP-6242E/GM	VP-5243E/GM
Overall arm length		210 (first arm) + 210 (second arm) = 420 mm	210 (first arm) + 220 (second arm) = 430 mm
Arm offset		J3 (front arm): 75 mm	-
Maximum motion area		R = 502 mm (end-effector mounting face) R = 432 mm (Point P: J4, J5, J6 center)	R = 500 mm (end-effector mounting face) R = 430 mm (Point P: J5, J6 center)
Motion range		J1 : ±160° J2 : ±120° J3 : +160°, +19° J4 : ±160° J5 : ±120° J6 : ±360°	J1 : ±160° J2 : ±120° J3 : +136°, -128° J4 : none J5 : ±120° J6 : ±360°
Maximum payload		2 kg (2.5 kg at wrist direction downward)	2.5 kg (3 kg at wrist direction downward)
Maximum composite speed		3900 mm/s (at the center of an end-effector mounting face)	
Position repeatability **(Note 3)**		In each of X, Y and Z directions: ±0.02 mm	
Maximum allowable inertia moment		Around J4 and J5: 0.030 kgm^2 Around J6: 0.007 kgm^2	Around J5: 0.040 kgm^2 Around J6: 0.010 kgm^2
Position detection		Absolute encoder	
Drive motor and brake		AC servomotors for all joints, Brakes for all joints	
User air piping		4 systems (ϕ4x4)	
User signal line		9 (for proximity sensor signals, etc.)	
Air source	Operating pressure	0.10MPa to 0.39MPa	
	Maximum allowable pressure	0.49MPa	
Degree of protection		IP30	
Airborne noise (A-weighted equivalent continuous sound pressure level)		80 dB or less	
Weight		Approx. 15 kg (32 lbs)	Approx. 13 kg (29 lbs)

Note 1: The 5-axis type robot has not the 4th axis (J4).

Note 2: The model name of robot set refers to the model name of a complete set including a robot unit and robot controller.

Note 3: Position repeatability is the value at constant ambient temperature.

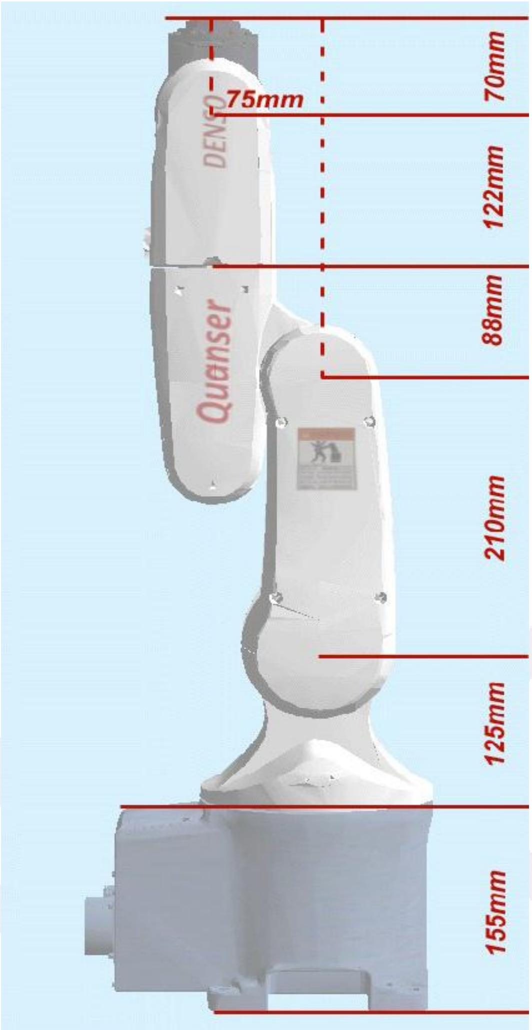


Figure A.2 Link lengths of the robot

The specifications of the Denso 6-Axis robot encoders and motors are summarized. The Denso robot arm has six encoders that measure the angular position of the six motors. The link lengths are displayed in millimeters in Figure A.2. World frame and join frames used for system are shown in Figure A.3. The encoders and motors specifications are summarized in Table A.3. This table lists the encoder resolution, motors gear ratios, motors torque constants, and joints hard stop limits. The joints 2, 3, and 5 are zero when the robot is completely straightened up as depicted in demonstrates of the world frame and the joint frames.

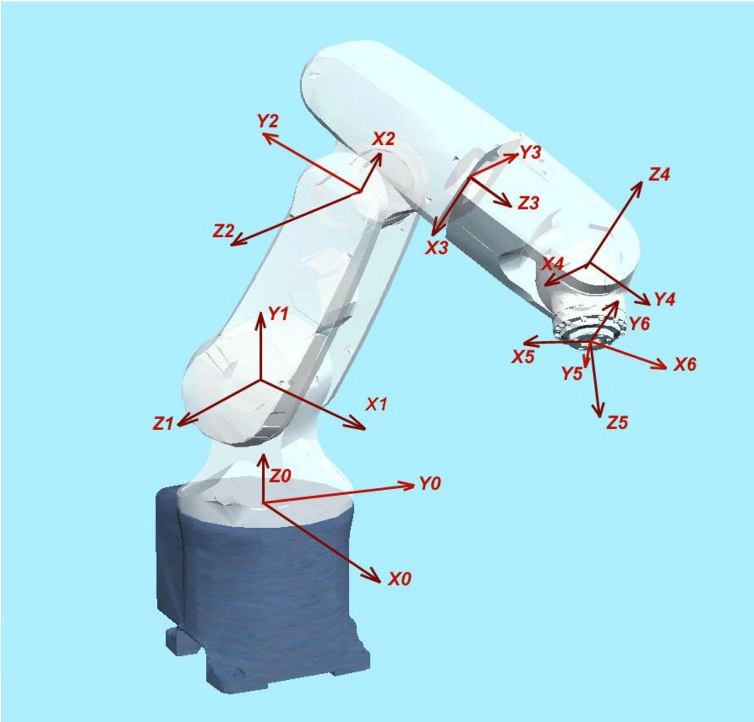


Figure A.3 World frame and join frames used for system.

Table A.3 Motor and encoder calibration specifications for Denso 6-Axis robot VP6242G

Motor/Encode Starting with Base	Motor Gear Ratio	Encoder Calibration (count/deg)	Torque Constant N.m/Amp	Joint maximum stop limit (deg)	Joint minimum stop limit (deg)
1	120:1	43690.666670	0.38	160	-160
2	160:1	58254.222220	0.38	120	-120
3	120:1	43690.666670	0.22	160	20
4	100:1	36408.888890	0.21	160	-160
5	100:1	36408.888890	0.21	120	-120
6	100:1	36408.888890	0.21	360	-360

A.4 Denso Robot Configuration

Caution: If the equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired. The system configuration is shown in Figure A.4.

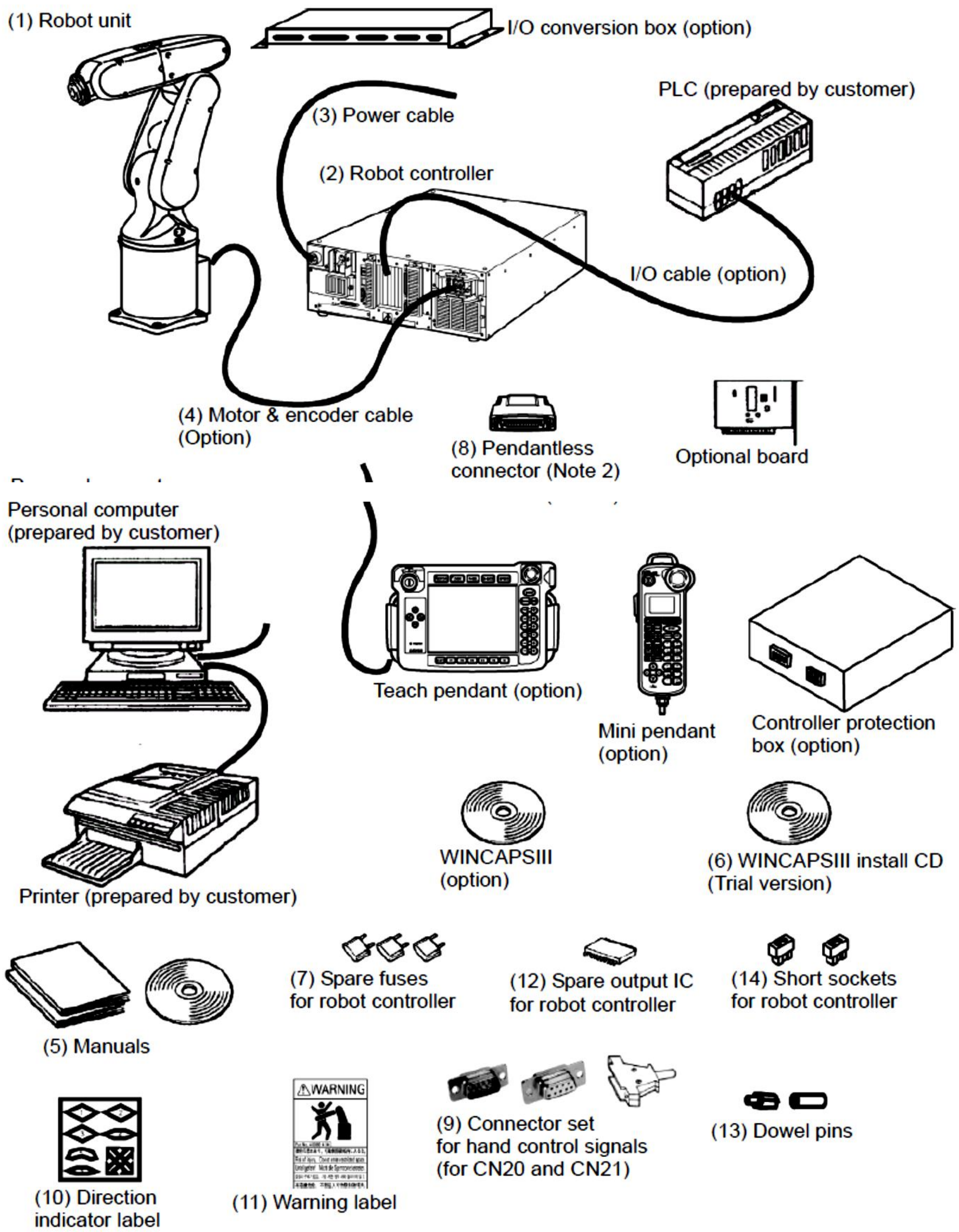


Figure A.4 System configuration [66]

A.5 Denso Robot Prerequisites

There are several Denso parts that need to be consulted prior to controlling the system in open-architecture mode. This document does not go through many basic robot operations that are covered in the Denso robot manuals. To successfully carry out this laboratory, the prerequisites are:

1. In order to install the Denso 6-Axis robot system the user must carefully study references 3, 4, and 5. This document goes though how to install, test, and wire the system for open-architecture implementation.

2. If the multi-axis ATI force/torque sensor with corresponding data acquisition board is used, make sure for installation and other information, described how to mount and wire the ATI F/T sensor on the Denso robot arm.
3. The user should be familiar with using QUARC to control and monitor the robot in real-time and to design a controller through Simulink.
4. The information for the serial port PCI card are provided.
5. The information for installation and communication with optional Linear Actuator is given.

A.6 Overview of Open Architecture Tele-operation System

In open-architecture mode, the encoder sensors as well as the actuators in the system are connected to the PC through the Quanser control module. This is shown in Figure A.5. In this mode, the controller algorithm is designed in Simulink and implemented with QUARC instead of using Denso own controller. QUARC generates real-time code from a Simulink model and targets it on the processor of a PC. The components involved in a single PC bilateral tele-operation setup are demonstrated.
Master robot (Quanser controller), slave robot (Denso 6-Axis robot), and force/torque sensor. The F/T sensor is mounted on the Denso robot end-effector and wired to the data acquisition board in the PC through the amplifier and special, optional cables. Quanser control module contains the amplifiers and embedded controllers for the Denso arm. It is connected to the specialized network card in the PC through Ethernet cable. The controller robot utilizes a cable to make this connection. In case a tool is delivered with the robot then the RS232 cables are required to wire the tool electronics box to the PC and the Denso robot back panel. It should be mentioned that the Quanser control module should preferably be connected to the specialized external network card in the PC. The components are involved in a double PC bilateral tele-operation. PC (Master PC) communicates with the master robot and transmits the commands to the Slave PC over the communication network. The slave PC (Denso PC), on the other side, communicates with the slave robot and measures the force/torque sensor data and sends the measurements back to the master PC over the communication network.

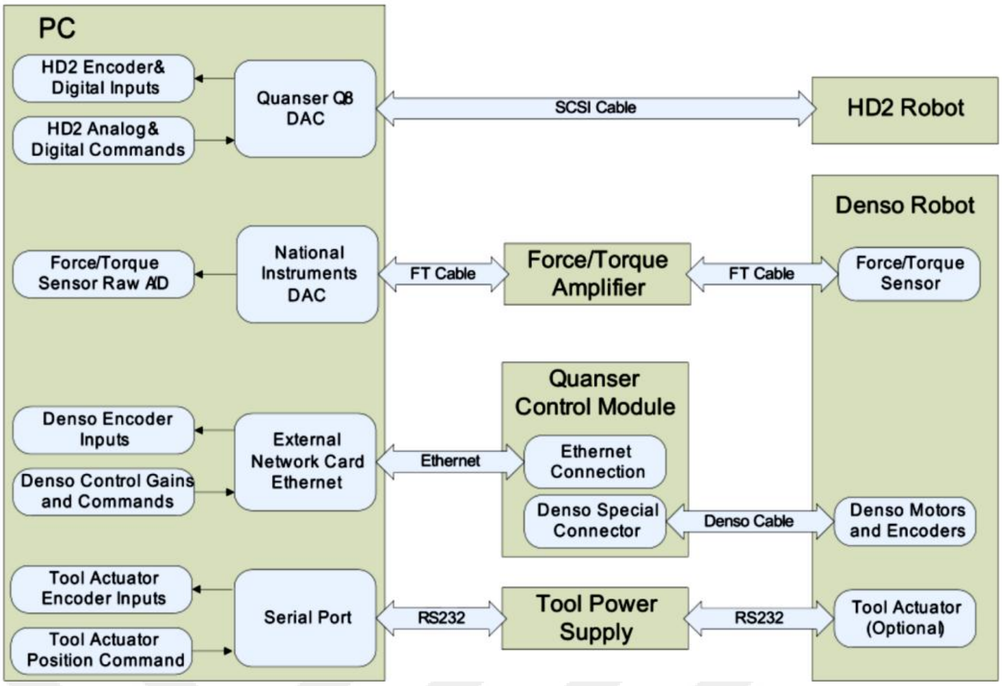


Figure A.5 Open architecture tele-operation system [66]

CURRICULUM VITAE

PERSONAL INFORMATION

Name and Surname: Ahmed Rahman Jasim ALMUSAWI

Nationality: Republic of Iraq

Birth place and Date: Iraq, 1979

Phone number: +905532822301

Email: engmktron@gmail.com

EDUCATION

	Graduate school	Year
Master	University of Baghdad/ Mechatronics Engineering	2009
Bachelor	University of Baghdad/ Mechatronics Engineering	2003
High School	Almarkazia High School	1998

WORKING EXPERIENCE

	Place	Enrollment
2003-2009	University of Baghdad/ Mechatronics Engineering	Instructor
2009-2012	University of Baghdad/ Mechatronics Engineering	Lecturer
2006-2012	Senior Technical Advisor	Alkhora Group
2003-2006	Technical Manager	Consultative Bureau Information System and Computer

PUBLICATIONS

1. Ahmed R. J. Almusawi, L. Canan Dülger, and Sadettin Kapucu. (2016). Haptik Teknoloji ile Robot Kolunun Denetimi ve Medikal Operasyonlarda Uygulanabilirliği, *Conference on Otomatik Kontrol Ulusal Toplantısı (TOK'2016)*.
2. Ahmed R. J. Almusawi, L. Canan Dülger, and Sadettin Kapucu. (2016). A new artificial neural network approach in solving inverse kinematics of robotic arm (Denso vp6242), *Computational Intelligence and Neuroscience*, 2016.
3. Ahmed R. J. Almusawi, L. Canan Dülger, and Sadettin Kapucu. (2016). Robotic Arm Dynamic and Simulation with Virtual Reality Model (VRM), *The 3rd International Conference on Control, Decision and Information Technologies (CoDIT'16)*, **3**, IEEE http://www.codit2016.com.
4. O. Yilmaz, Ahmed R. J. Almusawi, Adnan. A. Ugla, and O. Keskin. (2015). Design, Construction, and Controlling of A Shaped Metal Deposition Machine Using Arc Metal-Wire System, *8th International Conference and Exhibition on Design and Production of Machines and Dies/Molds*, **8**, http://diemold.net/Default.aspx.
5. Nabeel K Abid Al-Sahib, and Ahmed R. J. Almusawi. (2010). Guiding Mobile Robot by Applying Fuzzy Approach on Sonar Sensors, *Al-Khwarizmi Engineering Journal*, **6**, (3), 36-44.

Languages

Arabic, English, and Turkish.