UNIVERSITY OF PATRAS

DEPARTMENT OF MECHANICAL ENGINEERING AND AERONAUTICS

DIVISION OF DESIGN & MANUFACTURING

MACHINE DESIGN LABORATORY (MDL)

ROBOTICS GROUP

DIPLOMA THESIS

«FEA Simulation of hip fracture treated with rod-screw fixation and design of robotic surgery»

Bekas Andreas

1054470

Supervisors:

Nikolakopoulos Pantelis

Koustoumpardis Panagiotis

PATRAS, 6/2022

University of Patras, Department of Mechanical Engineering and Aeronautics
Bekas Andreas
© 2022 – All rights reserved

FEA Simulation of hip fracture treated with rod-screw fixation and design of robotic surgery

Bekas Andreas, 1054470

Supervisors:

Nikolakopoulos Pantelis

Koustoumpardis Panagiotis

ABSTRACT

It is an indisputable fact that surgical operations have evolved over the last years, though hip fracture rehabilitation is not always successful; many individuals experience pain, disability, and reduced quality of life afterward[1]. So, continuous emphasis should be placed on optimizing the surgical procedure and minimizing the risk of complications. Furthermore, Computer-Assisted Surgery (CAS) is an evolving method of integrating robotic technologies in the operating room, improving the operation's accuracy and efficiency[2]. So, this thesis is oriented towards evaluating a new way of Dynamic Hip Screw (DHS) insertion through the Ansys Finite Element Analysis (FEA) Software and then designing a manipulator's control system for its prosecution.

Rotation of femoral head fracture during the operation of DHS insertion due to screw's frictional forces is a subject of interest for surgeons because it leads to fault attachment of the two fragments. In the first part of this thesis, a new approach to surgical operation was proposed and studied through Ansys FEA software. Compared with the conventional way, this method's distinguishing feature is applying a traction force to the screw before its rotation, which causes the bone fractures to come in contact and develop frictional forces. Finally, the method's validity is evaluated based on the Ansys results, namely the sliding distance between the two fragments and the internal stresses of the bone.

The next step was to simulate the surgical operation utilizing a KUKA IIWA manipulator imported into Simulink, a block diagram environment software. The controller's development was based on kinematic and dynamic principles. Upon completion of the screw's finite element analysis, obtained values of forces and velocities, along with the screw's reaction forces during

tightening, were used for a more realistic simulation of Kuka’s operation. Finally, the applied joints' torque and end-effector's position and forces were assessed, confirming the system's validity, and highlighting its future perspectives to be integrated into an actual surgery room.

KEYWORDS

Computer-Assisted Surgery, Dynamic Hip Screw (DHS), Finite Element Analysis, Ansys Mechanical, Manipulator, Simulink, Force-Motion Hybrid Control Kuka IIWA.

Προσομοίωση με Πεπερασμένα Στοιχεία Σύνδεσης Κατάγματος Ισχίου με Κοχλίες και Σχεδιασμός Τροχιάς για Ρομποτική Επέμβαση

Μπέκας Ανδρέας, 1054470

Επιβλέποντες:

Νικολακόπουλος Παντελής

Κουστουμπάρδης Παναγιώτης

ΠΕΡΙΛΗΨΗ

Η αποκατάσταση κατάγματος ισχίου δεν είναι πάντα επιτυχής και έτσι πολλά άτομα βιώνουν πόνο, αναπηρία και μειωμένη ποιότητα ζωής στη μετέπειτα ζωή τους. Γι' αυτό, θα πρέπει να δίνεται συνεχής έμφαση στη βελτίωση της επέμβασης και στην ελαχιστοποίηση του ρίσκου επιπλοκών.

Η περιστροφή της κεφαλής του μηριαίου οστού κατά την σύσφιξη, λόγω των δυνάμεων τριβής που αναπτύσσονται ανάμεσα σε αυτήν και τον κοχλία, αποτελεί μια συχνή επιπλοκή, οδηγώντας σε ελαττωματική πώρωση των θραυσμάτων. Μια νέα προσέγγιση της χειρουργικής επέμβασης έχει προταθεί στη σπουδαστική εργασία του Μπέκα Ανδρέα με τίτλο «Σχεδιασμός και Προσομοίωση Εισαγωγής Δυναμικού Κοχλία Ισχίου με την Χρήση Πεπερασμένων Στοιχείων».

Η μέθοδος που προτείνεται αποτελείται από δύο στάδια, την εφαρμογή αρχικά μιας ελκτικής δύναμης στον κοχλία και στη συνέχεια μίας ταχύτητας περιστροφής αυτού. Η ειδοποιός διαφορά αυτής της μεθόδου είναι η ελκτική δύναμη, που οδηγεί τα οστά να έρθουν σε επαφή και να αναπτυχθούν δυνάμεις τριβής μεταξύ τους, που θα αποτρέψουν την περιστροφή. Με βάση την προσομοίωση της διαδικασίας, όπως ολοκληρώθηκε στο πρόγραμμα πεπερασμένων στοιχείων Ansys, προέκυψε ότι η περιστροφή της κεφαλής του μηριαίου οστού μπορεί να αποτραπεί με αυτήν την μέθοδο. Τέλος, η μέγιστη ελαστική πίεση που ασκήθηκε στα οστά ήταν αποδεκτή σύμφωνα με τα κριτήρια Von-Mises.

Με την ολοκλήρωση της ανάλυσης με πεπερασμένα στοιχεία, οι τιμές των δυνάμεων και των ταχυτήτων χρησιμοποιήθηκαν ως βάση για τον προγραμματισμό, μέσω Simulink, της επέμβασης με τη χρήση του ρομποτικού βραχίονα Kuka IIWA. Εφαρμόστηκε υβριδικός έλεγχος, με έλεγχο δύναμης για την έλξη και έλεγχο θέσης για τη γωνία περιστροφής. Επιπλέον, εξωτερικές δυνάμεις και ροπές εφαρμόστηκαν στο άκρο εργασίας του Kuka προσομοιώνοντας τις εξωτερικές δυνάμεις

που, σύμφωνα με το Ansys, θα εφαρμόζονταν στον κοχλία κατά την διαδικασία της σύσφιξης. Τα αποτελέσματα των ροπών στις αρθρώσεις και της θέσης του άκρου εργασίας αξιολογήθηκαν και συζητήθηκαν στο τέλος της εργασίας.

ΛΕΞΕΙΣ-ΚΛΕΙΔΙΑ

Δυναμικός Κοχλίας Ισχίου, Χειρουργική Ρομποτική, Μέθοδος Πεπερασμένων Στοιχείων, Ansys Mechanical, Ρομποτικός Βραχίονας, Υβριδικός έλεγχος Θέσης-Δύναμης, Simulink, Kuka IIWA.

ACKNOWLEDGEMENT

I would like to express my deep gratitude to all my professors so far, but especially to Professors Nikolakopoulos Pantelis and Koustoumpardis Panagiotis. They gave me the unique opportunity to start this wonderful project on the topic of surgical robotics. Firstly, it was an honor being supervised by Professor Nikolakopoulos Pantelis, that despite all the challenges, provided me with his valuable guidance and continuous encouragement throughout this project. I would also like to thank my co-supervisor, Koustoumpardis Panagiotis, for his constant help, advice, and all the resources he offered for the completion of the project. Also, I cannot fail to mention the Director of Orthopedic Surgery at General University Hospital of Patras, Mr. Kouzelis Antonios, who entrusted us with this research.

Finally, I would like to thank my parents, Bekas Panagiotis and Xexaki Evdoxia, my siblings, my uncles, my cousins, and my grandmother for believing in me and supporting every decision I make in every possible way.

Table of Contents

ABSTRACT ..... **v**

KEYWORDS ..... **vi**

ΠΕΡΙΛΗΨΗ ..... **vii**

ACKNOWLEDGEMENT ..... **ix**

LIST OF FIGURES **4**

LIST OF TABLES **7**

1. INTRODUCTION **8**

1.1. MOTIVATIONS 8

1.2. OBJECTIVES - THESIS OVERVIEW 9

2. LITERATURE REVIEW **11**

2.1. INTRODUCTION TO ROBOTS 11

2.2. COMPUTER-ASSISTED SURGERY (CAS) 12

2.2.1. HISTORY OF ROBOTICS IN SURGERY 12

2.2.2. APPLICATIONS 13

2.3. COMPUTER-ASSISTED ORTHOPEDIC SURGERY (CAOS) 15

2.3.1. ELEMENTS OF CAOS SYSTEMS 15

2.3.2. AUTONOMOUS SYSTEMS 15

2.3.3. PASSIVE SYSTEMS 16

2.3.4. SEMIAUTONOMOUS SYSTEMS 18

3. SURGICAL OPERATIVE TECHNIQUE **20**

3.1. PLACEMENT OF PATIENT AND PREPARATION OF HIP 20

3.2. SCREW INSERTION PROCESS 19

3.3. COMPLICATION PRESENTATION 20

4. FINITE ELEMENT ANALYSIS OF DYNAMIC HIP SCREW INSERTION **21**

4.1. DESCRIPTION OF THE MODEL 21
4.2. GEOMETRIC MODELLING 21
4.2.1. FEMUR GEOMETRY 21
4.2.2. SCREW GEOMETRY 24
4.2.3. MESHING & MATERIAL PROPERTIES 25
4.3. CONTACT MODELING 27
4.3.1. CONTACTS DETAILS 27
4.3.2. SCOPE 29
4.3.3. DEFINITION 29
4.3.4. ADVANCED 31
4.3.5. GEOMETRIC MODIFICATION 36
4.4. RESULTS 38
4.5. TRANSITION TO ROBOTICS 42
5. DESIGN OF DYNAMIC HIP SCREW INSERTION USING ROBOTIC ARM 44
5.1. INTRODUCTION 44
5.2. SIMULINK MODEL 44
5.2.1. KUKA DENAVIT-HARTENBERG EQUATIONS 44
5.2.2. KUKA MODEL 49
5.2.3. POSITION CONTROL 56
5.2.4. DIRECT FORCE CONTROL 60
5.2.5. HYBRID POSITION/FORCE CONTROL 63
5.2.6. FEEDFORWARD CONTROL 65
5.2.7. FORWARD KINEMATIC 66
5.2.8. MANIPULATOR FORCES 67
5.2.9. DIAGRAMS SUBSYSTEM 68

6. RESULTS 72

6.1. JOINTS TORQUES 72

6.2. END-EFFECTOR POSITION RESPONSE 76

6.3. END-EFFECTOR FORCE RESPONSE 79

7. CONCLUSIONS – DISCUSSION 81

BIBLIOGRAPHY 83

LIST OF FIGURES

FIGURE 2.1: TSOLUTION ONE® 16
FIGURE 2.2: IBLOCK 16
FIGURE 2.3: OPTOTRAK 3020 17
FIGURE 2.4:BRIGIT 17
FIGURE 2.5: ACROBOT 18
FIGURE 2.6:MAKO 18
FIGURE 2.7:NAVIO PFS 19
FIGURE 2.8: ROSA KNEE ROBOT 19
FIGURE 3.1: PATIENT POSITIONING 20
FIGURE 3.2: EXPERIMENTAL K-WIRE 21
FIGURE 3.3: GUIDE PIN MEASUREMENT 21
FIGURE 3.4: COMBINATION REAMER ASSEMBLY INSTRUCTIONS 21
FIGURE 3.5: LAG SCREW TAP ASSEMBLY 19
FIGURE 3.6: INSTRUMENT ASSEMBLY 19
FIGURE 3.7: FRACTURE COMPRESSION 20
FIGURE 4.1: AXIAL, CORONAL AND SAGITTAL PLANE OF FEMUR DICOM RESPECTIVELY 22
FIGURE 4.2:THRESHOLD RANGE OF HUMAN BONES 23
FIGURE 4.3: 3D-SLICER FEMUR DESIGN 23
FIGURE 4.4: FEMUR IMPORT TO CATIA 24
FIGURE 4.5: SCREW DESIGN 24
FIGURE 4.6: SCREW-FEMUR MODEL 25
FIGURE 4.7:MESH QUALITY 25
FIGURE 4.8: SKEWNESS MESH METRIC 26
FIGURE 4.9: FIXED CONDITION 26
FIGURE 4.10: 3D MODEL IN ANSYS 27
FIGURE 4.11: BONE-TO-BONE CONTACT 27
FIGURE 4.12: SCREW-TO-FEMORAL HEAD CONTACT 28
FIGURE 4.13: SCREW-TO-FEMUR CONTACT 28
FIGURE 4.14: CONTACT PARAMETERS 29
FIGURE 4.15: DEFINITION PARAMETERS 29
FIGURE 4.16: FRICTION COEFFICIENT PARAMETER 30
FIGURE 4.17: ADVANCED CONTACT PARAMETERS 32
FIGURE 4.18: PENETRATION REPRESENTATION 32

FIGURE 4.19: GEOMETRIC MODIFICATION PARAMETERS 36
FIGURE 4.20:INTERFACE TREATMENT PARAMETER 37
FIGURE 4.21:IMPLANT EQUIVALENT (VON-MISES) STRESS 38
FIGURE 4.22:SCREW DEFORMATION....38
FIGURE 4.23:FEMORAL HEAD DEFORMATION....39
FIGURE 4.24:FRAGMENT SLIDING DISTANCE 39
FIGURE 4.25:CONTACT FRICTIONAL STRESS....40
FIGURE 4.26:FEMORAL HEAD EQUIVALENT (VON-MISES) STRESS 41
FIGURE 4.27: SCREW'S JOINT TORQUES 42
FIGURE 4.28: TABULAR DATA OF MOMENTS....43
FIGURE 5.1: COORDINATE SYSTEMS [33]....45
FIGURE 5.2: HOMOGENEOUS MATRICES OF KUKA'S LINKS 47
FIGURE 5.3: KUKA IIWA SUBSYSTEM....49
FIGURE 5.4:KUKA MANIPULATOR....50
FIGURE 5.5: IIWA_LINK_4 BLOCK 51
FIGURE 5.6: SEVENTH JOINT LIMITS 52
FIGURE 5.7: EXTERNAL FORCES SUBSYSTEM 55
FIGURE 5.8: EXTERNAL FORCES DIAGRAM....55
FIGURE 5.9: MOMENTS IN BUILDING SIGNALS 56
FIGURE 5.10: POSITION CONTROL SUBSYSTEM 56
FIGURE 5.11: ROTATION FALSE SUBSYSTEM 57
FIGURE 5.12: ROTATION FALSE DIAGRAM....58
FIGURE 5.13: POSITION CONTROL BLOCK DIAGRAM 59
FIGURE 5.14: ROTATIONAL VELOCITY....60
FIGURE 5.15: FORCE CONTROL SUBSYSTEM 60
FIGURE 5.16: DIRECT FORCE CONTROL DIAGRAM 62
FIGURE 5.17: DESIRED FORCE SIGNAL....62
FIGURE 5.18: POSITION'S CONTROL MATRIX....63
FIGURE 5.19: FORCE'S CONTROL MATRIX....63
FIGURE 5.20: 5 DOF ← 6 DOF....64
FIGURE 5.21: 1 DOF ← 6 DOF....64
FIGURE 5.22: SYSTEM DIAGRAM....64
FIGURE 5.23: FEEDFORWARD SUBSYSTEM 65
FIGURE 5.24: FEEDFORWARD BLOCK'S DIAGRAM....65
FIGURE 5.25: FORWARD KINEMATICS SUBSYSTEM....66

FIGURE 5.26: FORWARD KINEMATICS DIAGRAM 66
FIGURE 5.27: MANIPULATOR'S FORCES CALCULATION BLOCK....67
FIGURE 5.28: FORCES CALCULATOR DIAGRAM....68
FIGURE 5.29: RESULTS SUBSYSTEM 68
FIGURE 5.30: END-EFFECTORS VELOCITIES DIAGRAM 69
FIGURE 6.1: JOINT1 TORQUE....73
FIGURE 6.2:JOINT2 TORQUE 73
FIGURE 6.3: JOINT3 TORQUE....73
FIGURE 6.4: JOINT4 TORQUE....74
FIGURE 6.5: JOINT5 TORQUE....74
FIGURE 6.6: JOINT6 TORQUE....74
FIGURE 6.7: JOINT7 TORQUE....75
FIGURE 6.8: DESIRED POSITION 76
FIGURE 6.9: REAL POSITION 76
FIGURE 6.10:DESIRED ANGLES 77
FIGURE 6.11: REAL ANGLES 77
FIGURE 6.12: POSITION ERROR....78
FIGURE 6.13: ROTATION ERROR 78
FIGURE 6.14: DESIRED FORCE 79
FIGURE 6.15: REAL FORCE 80
FIGURE 6.16: FORCES ERROR 80

LIST OF TABLES

TABLE 4.1: HOUNSFIELD THRESHOLD RANGE OF DIFFERENT HUMAN TISSUE [25] 22
TABLE 4.2: MATERIAL PROPERTIES 26
TABLE 4.3:BOUNDARY CONDITIONS 26
TABLE 5.1: KUKA D-H PARAMETERS [34] 46
TABLE 5.2: ROTATIONAL VELOCITY CALCULATION DIAGRAM 71
TABLE 5.3: ROTATIONAL VELOCITY TABLE 71
TABLE 6.1: JOINTS BOUND AND MAXIMUM TORQUES 75

1. INTRODUCTION

1.1. MOTIVATIONS

The Dynamic Hip Screw can be used as a fixation for the neck of femur fractures. Although the equipment comes with some manuals regarding the operative process and the proper handling of the tools, there are some blurry spots in operation. Some sub-procedures cause side effects that the surgeon comes to face, and many times he needs to improvise. The need to suggest thorough methods for the confrontation of such situations is imperative. This thesis seeks a specific solution for rotating the femoral head relative to the femur on the screw axis while tightening. The Director of the Orthopedic Surgery of General University Hospital of Patras, Mr. Kouzelis Antonios, observed this side-effect during the operation. For the settlement of the issue during the procedure, he observed that traction exerted on the screw afore the rotation could limit the relative rotation of the femoral head. So, the proof of this concept was confident to professors Nikolakopoulos Pantelis and Koustoumpardis Panagiotis of the Department of Mechanical Engineering & Aeronautics of the University of Patras. They, in turn, assigned the issue to the bachelor's thesis «Design and Simulation of the dynamic hip screw insertion with FEM».

Computer-Assisted Surgery (CAS), also known as Robotic Surgery, is a modern method of surgical operation characterized by its enhanced precision, flexibility, and control. By utilizing robotic systems, minimally invasive and orthopedic surgeries can occur, preventing patients from extended hospital stays, long recovery, large incisions, and scars. Patients experience less pain and discomfort while surgeons can perform more accurate operations with decreased risks of complications. Although surgeons must be trained in robot handling, results from using robots in operating rooms are encouraging. An increasing number of hospitals take advantage of medical robots; thus, the second part of this thesis is oriented towards fundamental principles of robotic arms. Fundamental kinematic and dynamic laws were utilized to introduce an innovative hybrid-control system, capable of managing the completion of the operation as performed by the FEA software. The aim of this autonomous system was that the Kuka IIWA robotic arm would operate the tightening of the dynamic hip screw with the implementation of dynamic and kinematic control simultaneously. The variables of this control are not arbitrary, but they came out of the Finite

Element Analysis, which proved that the selected values of forces and velocities are accepted. Matlab Simulink will execute all the control systems.

1.2. OBJECTIVES - THESIS OVERVIEW

The thesis contents are organized into seven chapters.

In **Chap. 1,** the motivation for the conduction of this study is reported, accompanied by some introductory information about the structure of the thesis.

In **Chapter 2,** the literature review of the Computer-Assisted Surgeries systems is presented. Initially, there is a summary of the history of robotic systems in the field of surgery. Then the emphasis is given to the area of Computer-Assisted Orthopedic Surgery.

Chapter 3 briefly describes the surgical operation as conducted by the surgeon. No emphasis has been placed on the surgical tools or the exact operation method of every step. The area of interest is in the second subchapter, which is devoted to the screw insertion.

Chapter 4 is devoted to the bachelor's thesis «Design and Simulation of the dynamic hip screw insertion with FEM», from which the necessary results were obtained for the design of the robotic control system. This thesis was about the simulation of the Dynamic Hip Screw (DHS) insertion with the usage of finite element analysis software. In the beginning, a general description of the desired model is presented; afterward, the steps for modeling the procedure are explained. Specifically, the way solid femur geometry came out of DICOM archives and the way the fractures have been assembled with the DHS. Also, actions necessary for completing the FEA simulation are presented, like meshing, boundary conditions, and contact details. Finally, the simulation results are analyzed about the integrity of the operation method, and the necessary data for the robotic simulation are extracted.

In **Chapters 5 and 6,** the generation of the controller and the simulator of the Kuka will be performed. Some information about the Kuka robot is presented, and some basic kinematic equations came from implementing the Denavit and Hartenberg method. What is presented then is the way that the manipulator's simulator was constructed through the SimpScape Multibody toolbox. After building the Kuka through the proper joints, links, inputs, and outputs, the rest of

the subchapters are devoted to designing the hybrid force/position control system. Each subchapter is dedicated to a specific subsystem of the Simulink, in which the corresponding theory is presented based on which the block is built. After the simulation has run, the resulting diagrams are plotted and analyzed about the model's validity and reliability.

Last, in **chapter 7**, conclusions and discussion about the research will be executed.

2. LITERATURE REVIEW

2.1. INTRODUCTION TO ROBOTS

According to the International Organization for Standardization [ISO], 2012, the term “industrial robot” is defined as «an **automatically controlled**, **reprogrammable**, **multipurpose manipulator programmable in three or more axis**, which can be either fixed in place or mobile for use in industrial automation applications».

The industrial robot is a programmable mechanical operator, able to move in different directions, equipped at its end with a working device called an action endpoint (or end-effector), and capable of performing various industrial tasks usually performed by the human.
A modern robot includes at least three basic building blocks:

1. the operator (arm, tool), which is the moving mechanical part,
2. the elements of action (motors, etc.), which activate the joints and
3. the computer that stores and executes work programs for the robot’s movements control.

The robot arm contains the main body and the wrist, which it has at the end of the tool (the final action element). The tool can be a welding head, a paint gun, a mechanical device, or a grab, which opens and closes depending on the application in which it is located to use the robot. Because all the tools are fastened at the end of robots, they are also called "end-effectors".

There are many kinds of classifications regarding robot applications. The main distinction of them is between the **industrial** and **service** robots. When referring to an **industrial** robot, it is commonly acceptable to talk about manipulators in manufacturing. They are usually fixed robots with a well-defined repetitive operation to manipulate in a strict pronounced environment. Industrial robots are ubiquitous in automotive industries’ production lines, such as car painting, welding, picking and place, and many more activities[3]. With the improvement of devices (sensors, end-effectors) and software, fixed robots started to impact operations with more human-robot interactions, such

as in medicine. On the other hand, **service** robots are usually considered to have human-robot interactions. They aim to assist humans in ordinary activities in a range of fields such as medicine, home, education, and defense [4].

The advantages of industrial robots can be summarized in high flexibility and productivity, better product quality, and improved quality of life afterward. So, like in many sectors of everyday life, such as manufacturing, transportation, and agriculture, computer-integrated systems have affected the field of medicine [2].

Medical Robotics has developed over the latter mainly decades, and the research done is concentrated in specific fields, from which the main ones are the following:

- Robotic Surgery,
- Rehabilitation Robotics,
- Virtual Reality Simulators with Visual Feedback - Power Simulation or without (Training Simulators with Haptic Display),
- Diagnostics Robotic Technology,
- Micro-Robotics [5].

2.2. COMPUTER-ASSISTED SURGERY (CAS)

2.2.1. HISTORY OF ROBOTICS IN SURGERY

The first robotic system implemented for surgical purposes was the PUMA 560 (Unimate, New Jersey, US) in the mid-1980. Its purpose was to raise the accuracy of the needle insertion during a 52-year-old male brain biopsy, which happened in conjunction with computed tomography (CT) guidance [6]. Many ideas about utilizing the technology in the name of robotic surgery have been proposed by Dr. Scott Fisher, a researcher at Ames Research Center of NASA, and Joseph Rosen, a plastic surgeon at the University of Stanford [7]. After one year of robotic-assisted surgery with the PUMA 560, the University of California, Devis initiated research projects to integrate robotic

systems into Total Hip Arthroplasty (THA). In May of 1990, a clinical test was performed on a dog [6]. The robot's name is ROBODOC, and it was used to prepare the femur bone for a THA. Simultaneously, neurosurgical robots were developed, with the Minerva System being introduced in 1990, while an assistive robotic arm for the positioning of laparoscopic camera called AESOP (Automated Endoscopic System for Optimal Positioning) made its appearance in 1993. As can be easily understood, computer-assisted surgery was initially developed for orthopedic and neurological surgeries. However, the most immediate and spontaneous integration of the CAS systems took place in urologic applications, and thus emphasis has been given to improving those systems [8].

2.2.2. APPLICATIONS

The purpose of CAS is to improve the treatment process by taking advantage of all the robotics capabilities available. First, patient data are extracted in various forms, such as images, genetic information, and lab results and after being combined with well-known information about human anatomy and physiology, a patient-specific model is constructed. Based on that “Virtual Reality” model, a plan is created for the surgical procedure. After the completion, the virtual interventional plan is registered on the real patient. During the whole procedure, information is generated for the intervention to be assessed with the goal of improving the procedure[2]. Some of the applications of robotic systems in surgery are:

Urology

Most radical prostatectomies in the United States are performed by a robot, specifically the Da Vinci system [9]. Integrating robotic-assisted systems in urology has resulted in less blood loss, shorter hospital stays, and fewer complications compared to conventional surgery [10]. Most of the urological cases in that robots are used concern the prostate disease. However, there is a valuable contribution of those systems in kidney and bladder diseases as well [8]

Cardiothoracic Surgery

One field of surgery in which robotic-assisted systems were implemented is coronary artery bypass grafting (CABG). It was first performed in the United States in 1999 [11]. After that, many cardiothoracic surgeries, like mitral valve replacement and myocardial tumor resection, have been facilitated integrating robotic systems in the surgery room [8].

Spine Surgery

Pedicle screw placement is a surgical operation that demands high accuracy. Tissues near the pedicle, like neurological deficits or vascular injuries, can easily be affected by incorrect placement of the screw [12]. For this reason, robotic assistance systems have been deployed throughout the time to facilitate the screw's insertion with high accuracy. Firstly, images are acquired preoperatively for the design of the desired planning. After registration of the robotic system on the patient, fluoroscopic images are used throughout the operation for the guidance of the pedicle screws in the predesigned path [13].

Orthopedic Surgery

Computer-assisted orthopedic surgery (CAOS) systems are a relatively new and emerging technology that focuses on assisting surgeries concerning musculoskeletal diseases and traumatology. Operations related to orthopedic surgery include fracture reduction and fixation, implant placement, ligament reconstruction, and drilling of bones. Robots' repeatability can increase the precision of the surgery, preventing postoperative and intraoperative complications [14]. CAOS systems include preoperative planning, manipulators or other robotic devices, navigators, and patient-specific surgical templates [15].

2.3. COMPUTER-ASSISTED ORTHOPEDIC SURGERY (CAOS)

2.3.1. ELEMENTS OF CAOS SYSTEMS

Surgeons have limited visibility of the operative tissue during minimally invasive surgeries. Modern technologies are used to encounter this limitation. Various devices are developed for the surgeons to obtain real-time visual feedback of the surgical tool on the tissue of interest. The function of the CAOS systems is based on the representation of the target object in a real-time display. The target organ or tissue combined with the represented plan is called a virtual object, and the link between those two, as said by Guoyan et al. [14], takes place through the navigator. Navigator uses a global coordinate system in which the target object and the surgical tool are represented based on their position and orientation. For the initiation of the procedure, calibration, registration, and referencing are needed to be done in advance. Calibration is the process of describing the tool's geometry and dimensions at the end-effector so that it can be precisely represented in the virtual environment. Registration is the process of providing the geometrical transformation between the surgical tool and the target tissue so that the relative position of the tool can be precisely determined. The third action for the CAOS to initiate is the referencing. This process is necessary to prevent deregulation of the system if the navigator or the target tissue moves.

2.3.2. AUTONOMOUS SYSTEMS

Autonomous systems, as their name suggests, operate autonomously without the interaction of the surgeon. The surgeon is responsible for the patient's correct dissection and the robotic system's initial set-up. Besides that, the surgeon plays the supervisor's role during the whole operation while the robotic system operates without an inter-operatively input. All the procedure has been designed during the pre-operation when the desired path of the end-effector of the robot has been planned [8]. The disadvantages of this type of system are the increase in the total time of the surgery due to registration delay and the pre-operational planning, as well as the incapability of the surgeon to interact inter-operatively.

TSolution One®

TSolution One® system, designed by the Think Surgical corporation, is evolution of ROBODOC and the first robotic system used for the Total Hip Arthroplasty (THA). Its purpose is to mill the cartilage and the bone during the intraoperative stage so that the implant can fit exactly. The procedure of milling the bone and positioning of the implant has been preoperatively planned at the workstation [16].



Figure 2.1: TSolution One®

iBlock

In 2010, FDA approved iBlock, designed by OMNIlife. An autonomous surgery robot that reduced the mean time of femoral resection compared to conventional methods and enhanced the efficacy of the TKA. After it is mounted on the patient's bone, and according to the surgeon's plan, an oscillating saw performs the resection[8].



Figure 2.2: iBlock

2.3.3. PASSIVE SYSTEMS

Unlike autonomous systems, passive systems depend only on the surgeon's moves. They provide the surgeon with the desired path the tool must pass through; passive systems are also called computer-assisted or navigation systems[17]. During the operation, the surgeon fully controls of the conventional tools without any limitation. At the same time, the robotic system only provides guidance and visualization of the surgical tool relative to the target organ [18]. There are three types of navigation depend on the way that images are obtained. The first type is obtaining volumetric images like CT, MRI, or ultrasound echograms. This is the most accurate navigation method, while the tool's path planning has been completed during the preoperative stage. The second one is the imageless navigation, which uses pointing landmarks and does not take account

of the uniqueness of each patient's anatomy. Kinematic information is obtained only intraoperatively, while no registration is required. The last one is fluoroscopic navigation which is preferred for trauma or spine surgeries. Images are obtained usually through C-arms during the operation, and despite the poor quality of the images, it is preferred due to the smaller dose of radiation [15].

Optotrak 3020

Optotrak 3020, designed by Northern Digital, Waterloo, Canada, is an optical tool designed for measurement of human motion. However, studies have been made with this in the fields of biomechanics and surgery. Its purpose is to locate and measure the position of infrared light-emitting diodes attached on tool of interest. This is accomplished through three cameras mounted on a bar calibrated by the manufacturer [19]

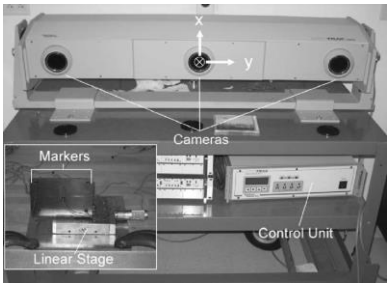


Figure 2.3: Optotrak 3020

BRIGIT

BRIGIT (Bone Resection Instrument Guidance by Intelligent Telemanipulator) is a robot mounted on a wheeled trolly developed by a French company, MedTech SA. Its purpose is to help surgeons with TKR and osteotomies. After the patient's dissection the bone is attached properly and through the end-effector of the robotic arm the registration takes place collecting anatomical landmarks. Next, a guiding tool is attached at the end-effector and locked at the appropriate position of the bone. Then, surgeon starts a free hand saw guided only mechanically by the guiding tool [6].



Figure 2.4:BRIGIT

2.3.4. SEMIAUTONOMOUS SYSTEMS

Semiautonomous systems are categorized somewhere between the autonomous and the passive systems. The surgeon moves the manipulator's end-effector in collaboration with the system's internal control, which limits some displacements within the desired boundary. Closed loop limits the tool's action either in terms of position or orientation or in terms of the speed of the cutting tool. With semiautonomous systems, the hand of the surgeon cannot deviate from the preoperatively planned path [8].

Acrobot

Acrobot can be said to be the evolution of the first robotic surgery system Puma 560. Although it was restricted from clinics, Brian Davies from Imperial College of Science, London, programmed it to operate autonomously against benign prostatic hyperplasia. Although initially found it efficient, Surgeons tended to desire a more human-robot collaborative system. So, the Active Constraint ROBOT (ACROBOT) was finally designed, constraining surgeon's moves in the preplanned area through haptic feedback. Its utility was limited for minimally invasive UKA [6].



Figure 2.5: Acrobot

MAKO

Mako is an FDA-approved semiautonomous tactile robotic system that uses images preoperatively for operation planning. During the operation, the surgeon fully controls the end-effector's position of the robotic arm. The plan is designed based on the patient's bone structure obtained through CT and the CAD model of the implant. Based on the plan, if the tool exceeds the boundaries of the area of interest, then the burr exerts resistance limiting the burr to be only within the accepted volume [6]. Throughout the operation, tactile feedback is continuously provided to the surgeon for extra safety [20].



Figure 2.6:MAKO

NAVIO PFS



Figure 2.7:Navio PFS

NAVIO Precision Free-Hand Sculptor (PFS) is an image-free semiautonomous robotic system designed by Smith & Nephew corporation and approved in 2012 by the FDA. Its purpose is to augment the UKA operation by constantly detecting the position of the burr and the patient's limb so that the accuracy is optimized [21]

ROSA Knee Robot

Although ROSA was initially used for cranial surgery, lately, Zimmer Biomet decided to adapt it to operate on TKA. It was first used in Australia in 2018, then spread to the United States, and in 2019 the FDA approved it [8]. Its design was based on experienced surgeons that had previously used similar surgical systems for knee arthroplasty. The purpose of ROSA is not to replace the surgeons but to assist them by planning the resection and positioning the jig in the right place. The ROSA system does not need preoperative imaging, as the 3D modeling is created exclusively by landmarks on the target cartilage and bone. During the operation, the mode of the robotic arm switches between automatic, collaborative, and static. Automatic mode is on when the end-effector is at a distance from the patient, while the collaborative mode is activated when it reaches the knee. Then the surgeon can exert gentle forces on the cutting tool to guide it to the desired points inside the planned area. After the proper position of the jig and the verification of it through the virtual model, the mode is turned to static, so the robot is locked in place, and the surgeon can cut the bone accurately [22].



Figure 2.8: ROSA Knee Robot

3. SURGICAL OPERATIVE TECHNIQUE

This chapter will present the procedure of Dynamic Hip Screw (DHS) insertion for the repair of intra-articular, pertrochanteric fractures, and femoral neck fractures based on Stryker's "Omega3 System" Operative Technique [23]. The main section of interest is when the screw is inserted, so a brief abstract of the preparation will be presented until the point of tightening. Then, in a second subsection, the procedure of our interest will be elaborated.

3.1. PLACEMENT OF PATIENT AND PREPARATION OF HIP

The patient is placed in a supine position on the operating table. The healthy foot is placed in a relaxed elevated position to allow imaging mechanisms (e.g., x-rays) to access the injured foot on the lateral and anteroposterior axis.

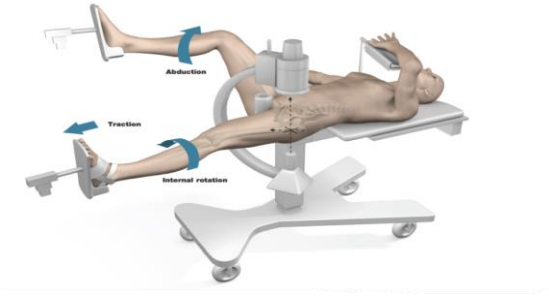


Figure 3.1: Patient Positioning

For the stabilization of the femur during the DHS insertion, a k-wire is used. Before inserting the guidewire inside the femur, a wire is experimentally placed on the outside of the bone. If the anteroposterior X-rays confirm that the placement of the wire and inclination is desirable, then the actual guidewire is placed in it simultaneously.

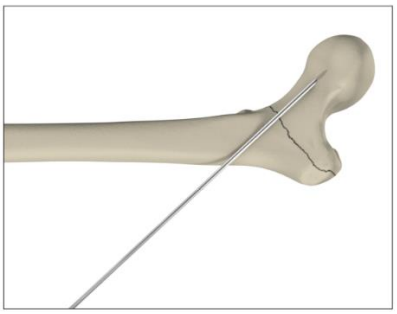


Figure 3.2: Experimental k-wire

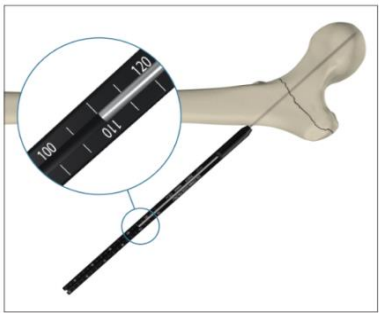


Figure 3.3: Guide Pin Measurement

The wire is inserted at the desired angle using the corresponding goniometer. It should be placed in the middle of the femoral head and at a depth up to the hypochondriac bone. Clinically the position of the wire guide is checked by anteroposterior and lateral radiographs.

Suppose the K-wire has penetrated up to the hypochondriac bone in order not to injure the cartilage and avoid a cut-out effect due to a short tip-apex distance [24]. In that case, the screw should have a measuring length of the measurement minus 10 cm.

Primary to the screw insertion, a hole is drilled with the help of a particular surgical drill. After connecting the hole drilling system and with the K-wire guide, the hole drilling starts.

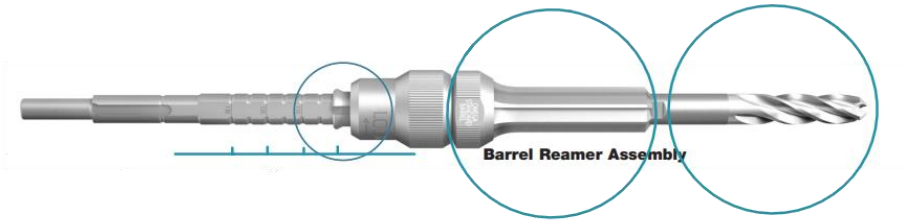


Figure 3.4: Combination Reamer Assembly Instructions

During the opening, breaks and spraying of the area are required to prevent bone necrosis due to high temperatures. The guidewire may be removed by completing the hole. Then, with the centering sleeve and a little pressure, it is repositioned. A hammer can be used to fasten it with soft blows if necessary.

Then the appropriate tools are used to create the thread of the potential screw. The thread is created in the previously formed hole and at the corresponding depth. Through the visualization

mechanisms, the depth is confirmed. It is recommended to apply this step only in the case of dense and healthy bone and to avoid it in cases of confirmed osteoporosis.

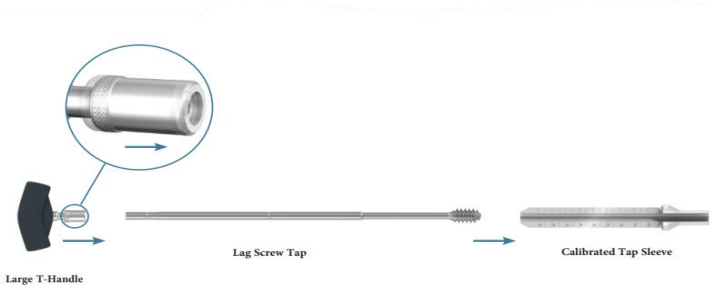


Figure 3.5: Lag Screw Tap Assembly

3.2. SCREW INSERTION PROCESS

The insertion process is the step that will be studied in this thesis and is about screw tightening. Instruments used to tighten the DHS screw include the **Lag Screw** itself, the DHS **Insertion Wrench**, the DHS **Plate,** and their **Insertion Sleeve**. The screw is first connected to the wrench to assemble the system and tightened to it through a particular slot to avoid possible damage. The wrench is then inserted into the plate hole to slide into it. Finally, the centering sleeve is placed on the wrench, and after the assembled system is slid on the guidewire and the sleeve is applied to the hole, the tightening process begins.

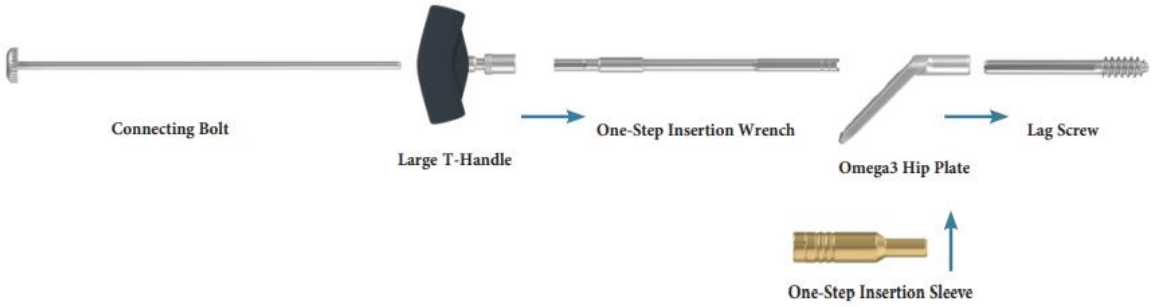


Figure 3.6: Instrument Assembly

The screw is inserted to the desired depth, which can be detected from the scale on the key. Then, the centering sleeve is removed, and the DHS plate is driven into the hole with a special hammer. The wrench and wire are removed when the screw and plate installation is complete.



Figure 3.7: Fracture Compression

3.3. COMPLICATION PRESENTATION

During the DHS tightening process, practical difficulties presented, which concern the femoral head's rotation. As a result of the rotation, the femur head heals at an incorrect angle, causing issues with the kinematics of the patient's foot. It is therefore critical to design a method of tightening the dynamic screw with a focus on preventing rotation of the femoral head about the screw's axis.

In the next chapter a new method for tightening the surgical screw is proposed, which concerns its rotation after traction has been applied to the screw. In this way, the fragments come in contact with each other, and the frictional forces that are developed tend to prevent femur head relative rotation.

4. FINITE ELEMENT ANALYSIS OF DYNAMIC HIP SCREW INSERTION

4.1. DESCRIPTION OF THE MODEL

The "Screw Insertion Process" will be simulated in this chapter using the Finite Element Analysis Program "Ansys". The subchapters summarize Batchelor's Thesis, “Design and Simulation of the dynamic hip screw insertion with FEM”.

For the simulation, a total time of 1.3 seconds was divided into three intervals of 0.2, 1.0, and 0.1 seconds, and three boundary conditions were applied. Initially, all surface nodes at the base of the femur were fixed for the whole time. The traction force increased from 0 to 8 Newtons during the first interval and remained constant during the other two intervals. The third limit condition concerns the screw's rotational speed, which developed in the second interval from 0 to 1 rad / s while reamaining constant in the third interval.

4.2. GEOMETRIC MODELLING

4.2.1. FEMUR GEOMETRY

3D-Slicer and Catia V5 were used for modeling the femur.

In medicine, there are modern techniques to render images of the human body of each patient. The most well-known and widely used are **magnetic resonance imaging (MRI), computed tomography (CT), positron emission tomography (PET)** and **single-photon emission computed tomography (SPECT)**.

CT scans of a patient were used to best approximate the geometry of the femur. These enter the 3D-Slicer in the form of digital image **DICOM (Digital Imaging and Communications in Medicine),** where they are presented in three planes:

1. the **coronal** plane,

2. the **sagittal** plane,
3. and the **axial** plane.



Figure 4.1: Axial, Coronal and Sagittal plane of femur DICOM respectively

Images are made up of cells called **pixels** and can have any shade of gray, from white to black, depending on the tissue density in the corresponding pixel. Tissue density can be assigned based on the **Hounsfield linear scale**. Based on this scale, water has zero value; elements with density higher than water have a positive value, while elements with a density lower than water have a negative value. The ratio that gives us the exact value is:

$$HU = 1000 * \frac{\mu tissue - \mu H2O}{\mu H2O} \quad 4.1$$

Where μ_{tissue}, is the linear tissue absorption coefficient.

Predefined threshold value	**Minimum**	**Maximum**
Bone (CT)	226	3071
Soft Tissue (CT)	-700	225
Muscle Tissue (CT)	-25	139
Fat Tissue (CT)	-205	-51
Skin Tissue (CT)	-766	-177

Table 4.1: Hounsfield Threshold Range of different human tissue [25]

Thus, as the density of the tissue increases, the HU value increases, and the color of the corresponding pixel acquires a lighter hue. Based on the table below, we define in 3D-Slicer the values of HU for the isolation of bone tissue, i.e., a range of values from 226 to 3071.

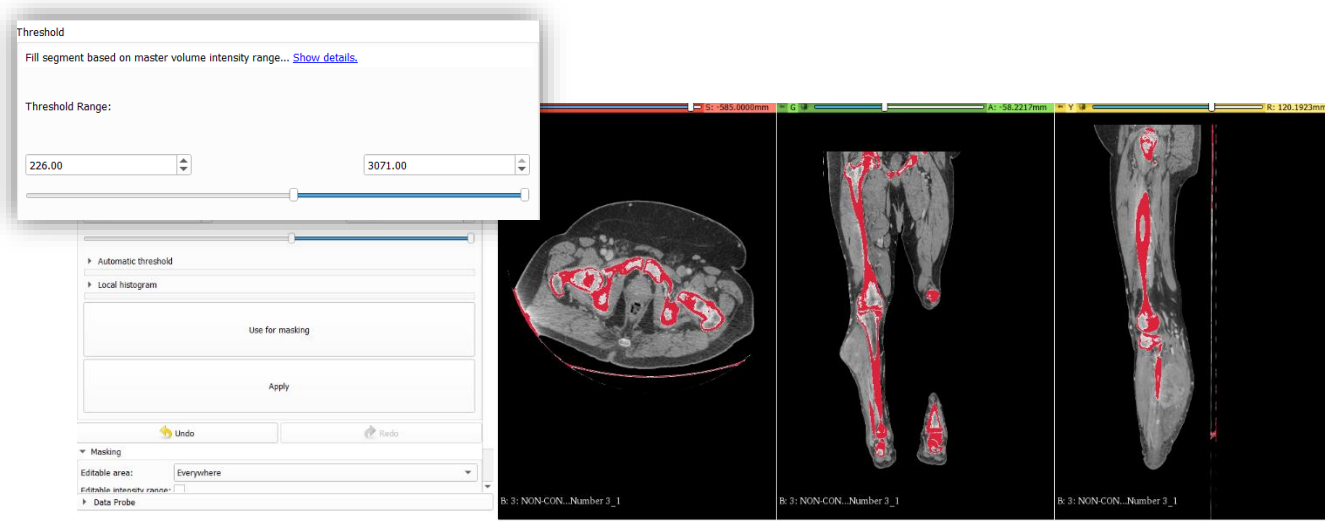


Figure 4.2:Threshold Range of Human Bones

Then with proper processing of the partitions through the commands **Smooth**, **Fill holes** and **Remove Extrusions,** and other options provided by the **Segment Editor** of 3D-Slicer to create the exact model we want, we get the part of the femur that interests us as shown below.

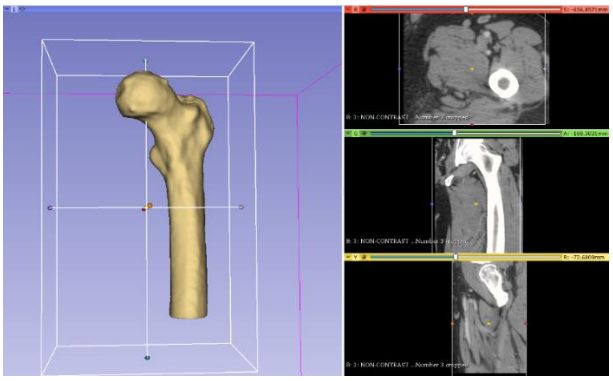


Figure 4.3: 3d-Slicer Femur Design

After the completion of the creation of the three-dimensional solid from the tomographs, the geometry is now extracted as a model through Segmentation and stored in the form of **Stereolithography (STL)**.

To complete the model and connect the femur to the screw, we insert the STL file in Catia. Due to the information contained in this file, i.e., that it transfers only surfaces and not tumors, small processing is needed to obtain the three-dimensional solid body of the hip.

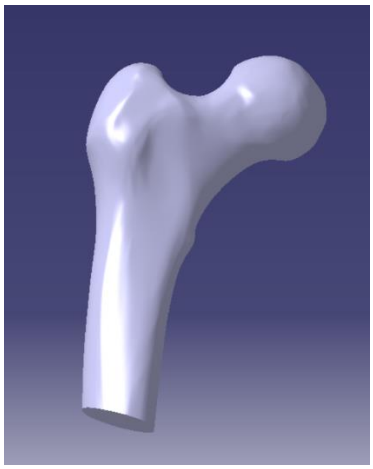


Figure 4.4: Femur Import to CATIA

4.2.2. SCREW GEOMETRY

The modeling of the screw potential was done using **Catia Student Edition**. Information was collected on the geometry of the screw, i.e., its length, diameter, length of the thread, the inner and outer diameter of the thread, and pitch. Thus, the model of the dynamic screw was created with a model of the "**Lag Screw**" screw of the **Gamma3** system of **Stryker Corporation**.

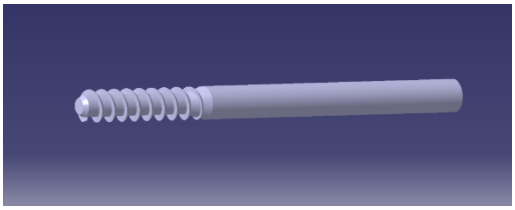


Figure 4.5: Screw Design

The completion of the design of the parts (Parts) follows their connection and the creation of the fracture in the femoral neck using the **split** command of Catia in **Part Design**. By placing the screw in the correct position according to the Technique of the Operation, we use the **Boolean Remove** command to create the space inside the bones where the screw will enter. Thus, we get the final shape-geometry of the model ready to be introduced in the **Ansys Workbench**.

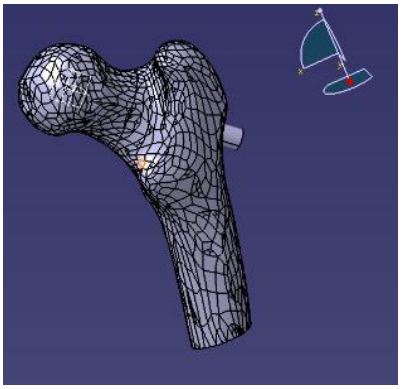


Figure 4.6: Screw-Femur Model

4.2.3. MESHING & MATERIAL PROPERTIES

Each part of the assembly has been meshed by solid187, a higher-order 3-D, 10-node element, that can be well suited to various CAD/CAM parts with irregular geometries. The total model femur-screw had a total of 73519 Nodes and 18417 Elements. The evaluation of the mesh quality will be based on the 'Skewness' value. Skewness directly shows how much the mesh structure is close to the ideal shape. The lower the value, the lower the divergence from the ideal. An average mesh Skewness below 0.94 is acceptable. The Average Skewness is 0.54, classifying our model mesh quality as "Good".

Mesh Metric	Skewness
Min	7.1397e-004
Max	0.9996
Average	0.54026
Standard Deviation	0.21963

Figure 4.7:Mesh Quality

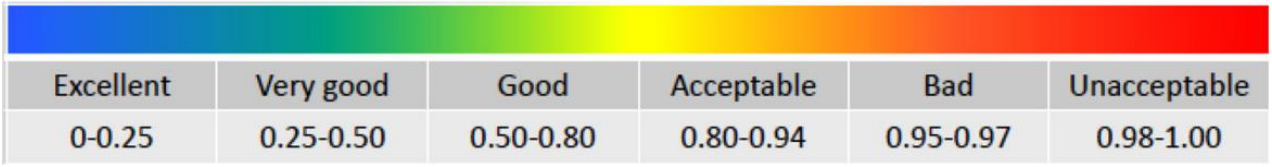


Figure 4.8: Skewness Mesh Metric

For modeling, it was assumed that there was no difference between cortical and trabecular bone. In addition, bones and screws were all continuous, isotropic, and linear materials. Their mechanical properties are summarized as shown in the table below.

PARAMETERS	**BONE**	**T-6AL-4V**
DENSITY (g/cm^3)	0.31	4.405
YOUNG'S MODULUS (MPa)	77	107000
POISSON'S RATIO	0.3	0.323

Table 4.2: Material Properties

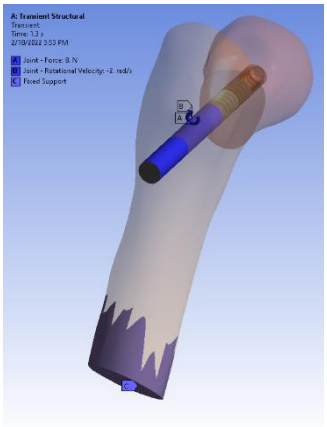


Figure 4.9: Fixed Condition

Surface nodes at the lower part of the lower femur were fixed in all degrees of freedom as a boundary condition. The whole simulation lasts 1.3 seconds and is divided into three steps. The load condition consisted of a traction force of 8 Newton applied at the top of the screw at its longitudinal axis at the first step. The rotational velocity of 2 rad/s was programmed to take place after the full implementation of the force while the acceleration to reach this velocity lasted 1 second. Details about boundary conditions are listed below.

STEPS	**TIME [S]**	**FORCE [N]**	**VELOCITY [RAD/S]**
1	0.	0.	0.
1	0.2	8.	0.
2	1.2	8.	-2.
3	1.3	8.	-2.

Table 4.3:Boundary Conditions

4.3. CONTACT MODELING

4.3.1. CONTACTS DETAILS

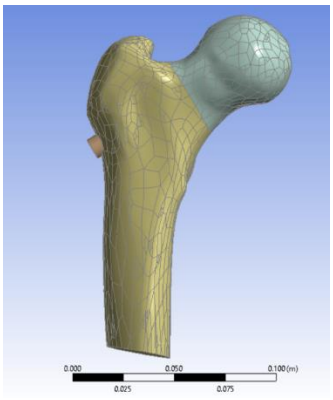


Figure 4.10: 3D Model in Ansys

Ansys' contact function enables the model to recognize that two or more bodies are in contact with each other. The choices given are either through solids, surfaces, or even lines between them. In the model of connection of the screw with the two fragments of the femur we find:

1) A contact between the two bones around the fracture.

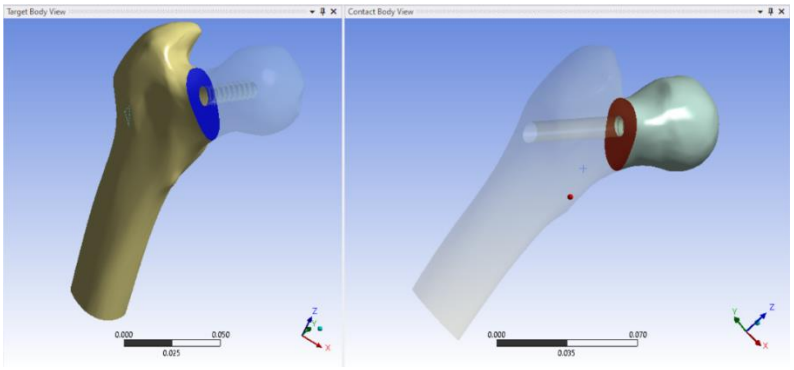


Figure 4.11: Bone-to-Bone Contact

2) A contact between the screw and the femoral head bone.

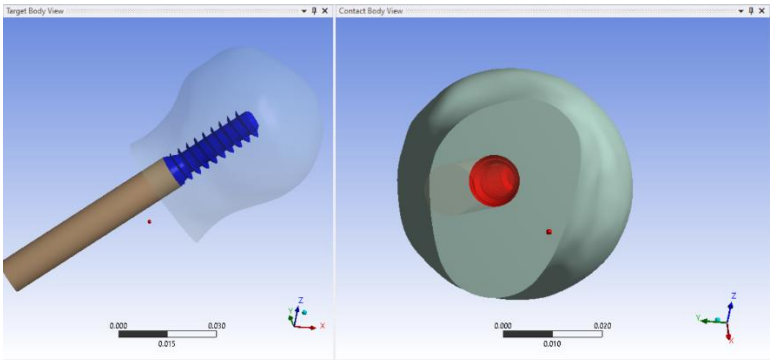


Figure 4.12: Screw-to-Femoral Head Contact

3) A contact between the screw and the femur.

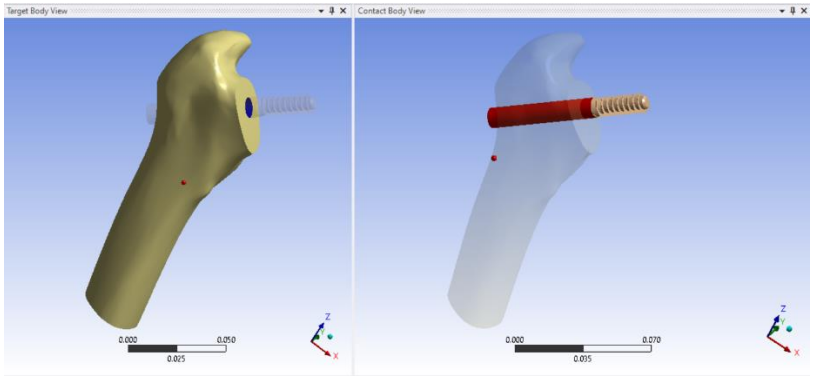


Figure 4.13: Screw-to-Femur Contact

The definition of these contacts prevents the surfaces of the three bodies from penetrating each other and creates the conditions for the development of surface **vertical forces** and **tangential frictional forces**.

As will be analyzed below, the parameters for completing the modeling of contacts are divided into four major categories:

Scope

Definition

Advanced

Geometric Modification

Details of "Bonded - No Selection To No Selection"

Scope	
Scoping Method	Geometry Selection
Contact	No Selection
Target	No Selection
Contact Bodies	No Selection
Target Bodies	No Selection
Protected	No
Definition	
Type	Bonded
Scope Mode	Manual
Behavior	Program Controlled
Trim Contact	Program Controlled
Suppressed	No
Advanced	
Formulation	Program Controlled
Small Sliding	Program Controlled
Detection Method	Program Controlled
Penetration Tolerance	Program Controlled
Elastic Slip Tolerance	Program Controlled
Normal Stiffness	Program Controlled
Update Stiffness	Program Controlled
Pinball Region	Program Controlled

Figure 4.14: Contact Parameters

4.3.2. SCOPE

The column under "**Scope**" mainly concerns the selection and display of the two interacting surfaces, **Contact** and **Target**. In the contact between the two bone fragments, it does not matter what the definition of each surface will be, as we will explain in the section "Definition-Behavior". While in the case of contact between screw and femur, although it will be defined which surface will be Contact and which Target based on Auto-Asymmetric, we will define as Contact the surface with the least stiffness, i.e., the surface of the bone. The screw surface will be set as Target.

4.3.3. DEFINITION

The options are given under "**Definition**" are the most used and relate to the following as shown in the image:

Type

Scope Mode

Behavior

Trim Contact

Suppressed

Definition	
Type	Bonded
Scope Mode	Manual
Behavior	Program Controlled
Trim Contact	Program Controlled
Suppressed	No

Figure 4.15: Definition Parameters

Type (& Friction Coefficient)

In the screw-femur model, the response of the system to external forces exerted on the screw is studied. However, they create surface forces (vertical and tangential) between the screw and the two bones. On some surfaces, the tangents can overcome the static friction to create relative movement (femur-screw), in other surfaces however we want the friction to prevent relative movement (bone-bone). To be able to study all this behavior we use the **Frictional** option in the **Type** of contacts. With this option, the contact bodies are charged tangentially up to a value from which they then begin to slide together. From the literature, the friction coefficient between the bone fragments is taken 0.46 while the frictional coefficient between the bones and the implant 0.3 [26], [27]. Also, M. Goffin et al., Jia Li et al. and others used the same coefficients at their computational study [28]–[30].

Definition	
Type	Frictional
Friction Coefficient	0.
Scope Mode	Manual
Behavior	Symmetric
Trim Contact	Program Controlled
Suppressed	No

Figure 4.16: Friction Coefficient Parameter

Scope Mode

It cannot be modified and simply states how the contact was created. Whether it is **Manual** or **Automatic**.

Behavior

The next parameter that can be configured is the behavior of the two surfaces **(Behavior)** based on which it will be determined whether it will be **symmetric** or **asymmetric**. In the case of **asymmetric contact**, all Contact elements are on one surface while all Target elements are on the other. Thus, the nodes of the Contact surface are restricted from penetrating the surfaces of the Target Elements. In the case of **symmetric contact**, the same applies to Target nodes on Contact surfaces.

The best way to model Surface-to-Surface contact is asymmetric behavior. Thus, in the case of the interface between the screw and the femur, there is a critical difference that can help to distinguish

the Contact and Target surface from each other, and this is the rigidity. Usually when a body is harder its surface is chosen as Target while the most flexible as Contact. In terms of elasticity, there is a difference between the two bone and metallic materials.

However, instead of choosing the **Asymmetric** behavior that in our judgment sets the behavior of the two surfaces, we choose the **Program Controlled** according to which **Auto Asymmetric** is activated for Flexible-Flexible bodies. Thus, the program selects the two surfaces on its own, regardless of the way we put them in the Scope.

However, in cases where the distinction between the two contacts (Target and Contact) is not clear, the **symmetric** contact applies. In this case, we have in contact between the two bones where both surfaces are flat, have the same hardness, have where we set the **Behavior** as **Symmetric**.

Trim Contact

The **Trim Contact** option can speed up the resolution time, as it reduces the number of contacts sent for resolution. This is done by calculating the proximity of the Contact nodes to the Tangent surfaces according to a Tolerance value set by the user in the case of **On**. The default in **Trim Contact** is **Program Controlled** which is usually set to **On**, except when the creation of contacts becomes Manual. In our case, the contacts were not created automatically, so the value is set to **Off** automatically.

Suppressed

The Suppressed option determines whether this contact will be included in the solution. Set to **On**.

4.3.4. ADVANCED

The column under the Advanced category provides options for modifying the following options:

Formulation

Small sliding

Detection Method

Penetration Tolerance

Elastic Slip Tolerance

Normal Stiffness

Update Stiffness

Stabilization Damping Factor

Pinball Region

Time Step Controls

Advanced	
Formulation	Program Controlled
Small Sliding	Program Controlled
Detection Method	Program Controlled
Penetration Tolerance	Program Controlled
Elastic Slip Tolerance	Program Controlled
Normal Stiffness	Program Controlled
Update Stiffness	Program Controlled
Stabilization Damping Factor	0.
Pinball Region	Program Controlled
Time Step Controls	None

Figure 4.17: Advanced Contact Parameters

Formulation

Ansys provides various models to define the contact between two surfaces. These can be one of the following four:

MPC (Multi-Point Constraint)

This method of contact modeling, although it can be used in other types of contacts, is recommended for linear contact types, i.e., either **Bonded** or **No Separation.**

Pure Penalty

The Pure Penalty modeling method allows the contact node to penetrate the target surface. Whenever there is vertical penetration of the node, a force tends to "push" it out in moderation:

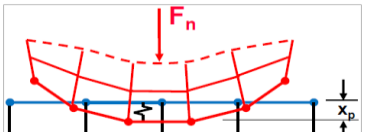


Figure 4.18: Penetration Representation

$$Fn = kn * xn \qquad 4.2$$

Where x_n indicates the vertical penetration of the node on the surface while the constant k_n is called the penetration hardness (Normal Stiffness). This parameter does not have any physical significance since the materials do not penetrate each other. Penetration hardness, however, plays

an important role in conveying the problem. When the system does not converge, reducing k_n can help decisively, however, the results will not be very realistic.

The same equation is used in the case that the tangential motion between the two surfaces is prevented:

$$Ft = kt * xt \quad (4.3)$$

Where Ft is the tangential force that tends to return the node tangentially to its original position, x_t is the tangential displacement and kt is the tangential stiffness.

In the case of **Normal Lagrange** and **Augmented Lagrange,** the method changes the vertical direction only. The **Pure Penalty** method is applied in the tangential direction.

Normal Lagrange

The Normal Lagrange method adds an extra degree of freedom, the contact pressure. When the point of contact is in contact with the target surface, the pressure is calculated according to the conditions that will prevent the bodies from penetrating each other.

$$Fn = \lambda \quad (4.4)$$

Where λ is the contact pressure or as it is called Lagrange multiplier.

Augmented Lagrange

The Augmented Lagrange method is a combination of the above two methods **Pure Penalty** and **Normal Lagrange**.

$$Fn = kn * xn + \lambda \quad (4.5)$$

Due to the Lagrange λ multiplier, this method is less sensitive to the penetration hardness k_n. The Augmented Lagrange method is generally recommended for slip problems (Frictional and Frictionless).

In our case, we leave the default **Program Controlled** which automatically selects the **Pure Penalty** method for the contact between rigid bodies and **Augmented Lagrange** for all other

cases. In the model with the bones and the screw, the methods in all three cases will be **Augmented, Lagrange**.

Small sliding

The **Small-Sliding** option in the case of relative movement between the screw and the bones must be removed. It is generally used in cases where the slip between the surfaces is small, i.e., less than about **20% of the contact length**. Then, the convergence speed of the model increases and shows better results.

In the case of our model, we want to achieve tightening of the screw, a movement that will cause a large relative slip of the screw surfaces with the bones. Also, the slip on the surface of the bones between them remains unknown and under clarification, so we should allow large slips between them. Therefore, in all three contacts, we set the option of **Small Sliding** to **Off**.

Penetration Tolerance

Activating the **Penetration Tolerance** option opens another parameter in the Advanced column, the Value or Factor, from which the depth at which the nodes of the Contact surface can penetrate the Target surface can be adjusted. Selecting the Value directly enters the maximum penetration depth while through the **Factor** value a factor from 0 to 1 is selected by the user.

In the case of **Program Controlled**, which we apply to our model in all three contacts, the penetration tolerance (Penetration Tolerance) is automatically calculated by the program.

Elastic Slip Tolerance

In case the method is set as Bonded, Frictional or Rough, the choice is given by the program of adjustment of the **Elastic Slip Tolerance**. This can be done by using **Value** in units of length or by using a **Factor**. Leaving it in the **Program Controlled** default, Ansys automatically sets the value of Tangential Contact Stiffness according to the value of **Normal Contact Stiffness** and the

properties of the material. On all three contacts of the screw-femur model, we leave the **Elastic Slip Tolerance** in the **Program Controlled**.

Normal Stiffness & Update Stiffness

As we analyzed in the paragraph "Formulation", the contacts between two surfaces in the case of the Augmented Lagrange method (as well as Pure Penalty) are characterized by the parameter of penetration hardness k_n.

The kn value can be set by the user in the **Normal Stiffness** definition. In this case, the **Manual** option is selected and the value of the factor that the user considers -from 0.01 to 10- is entered in the corresponding position (**Normal Stiffness Factor**). Otherwise, the default **Program Controlled** is left and the program calculates the constant according to the properties of the materials via **Mechanical APDL**. In all three contacts of the model, the **Program Controlled** in **Normal Stiffness** will be used.

Update Stiffness allows the program to modify the hardness based on the physics of the model. Thus, if it is observed at the end of an iteration that the penetration is large, the k_n coefficient increases, while if it is difficult to converge due to increased k_n, it can be reduced.

The options given are **Never**, **Each Iteration**, and **Each Iteration Aggressive**. Leaving the **Program Controlled** default and only if **Add Offset** has been set in the Interface Treatment, **Ramped Effects** is automatically selected as **Never**. Otherwise with **Program Controlled** each Iteration is activated automatically based on which at the end of each iteration the **Normal Stiffness** factor is renewed to get good results.

Stabilization Damping Factor

Used when the initial state between two surfaces appears as a gap as an error in creating the mesh of the original geometry. If the program does not realize that the actual state of the bodies is in contact, then the movement of the bodies may result. To avoid this, a **Stabilization Damping Factor** is set where it adds resistance to the relative motion of the bodies. This damping is added at each step where the state of the bodies is detected as open.

In our case **Stabilization Damping Factor** is left **zero** (0) as the default program has where the depreciation activation occurs only in the initial step and in no other.

Pinball Region

It is a spherical area, hence the name **Pinball Region**, with centers at the nodes of the Contact surface. If there are nodes in the sphere from the Target surface, then it is considered by the program that the nodes are close to each other and are monitored for whether they are really in contact. If indeed they are, then the program continues in their more detailed study of whether they slip, split, or penetrate. If the contact is not defined automatically, the only option given beyond Program Controlled is the **Radius**, in which the user sets the radius of the sphere. On the contrary, with the default **Program Controlled**, which we will use in all three contacts, the radius is calculated by the program.

Time Step Controls

The options given for **Time Step Controls** are **None**, **Automatic Bisection**, **Predict for Impact**. If in a step (sub-step) a large penetration happens, or the contact status is changed (Contact Status) and the **Automatic Bisection** is set then the program re-evaluates the step-through its bisection. The Predict for Impact option is recommended for models who expect a collision between bodies. The default None that will be used in our model does not affect the handling of the steps (Step Controls). It can also be used without fear if small steps are allowed in automatic time stepping.

4.3.5. GEOMETRIC MODIFICATION

The **Geometric Modification** list allows you to modify the following three parameters:

Interface Treatment

Contact Geometry Correction

Target Geometry Correction

Geometric Modification	
Interface Treatment	Adjust to Touch
Contact Geometry Correction	None
Target Geometry Correction	None

Figure 4.19: Geometric Modification Parameters

Interface Treatment

From these parameters, the **Interface Treatment** is activated due to the type of contact that has been defined, i.e., the friction contact (Frictional). This command allows you to select the initial state of the two contact surfaces. Thus, in all three contact cases **Adjust to Touch** is selected so that any initial gap between the surfaces is closed and any initial penetration is ignored. This creates a situation where there is zero initial stress on the surfaces.

Geometric Modification	
Interface Treatment	Adjust to Touch
Contact Geometry Correction	None
Target Geometry Correction	None

Figure 4.20:Interface Treatment Parameter

Geometry Correction

Contact Geometry Correction and Target Geometry Correction are left to None. The only thing they could provide in our case would be surface smoothing (Contact / Target Smoothing), i.e., the reduction of inaccuracies in the curved surfaces due to the local elements (Faceted elements). However, this has already been achieved to a large extent due to the high quality of the mesh (Fine Mesh).

4.4. RESULTS

MAXIMUM VON MISES STRESS DISTRIBUTION OF THE IMPLANT

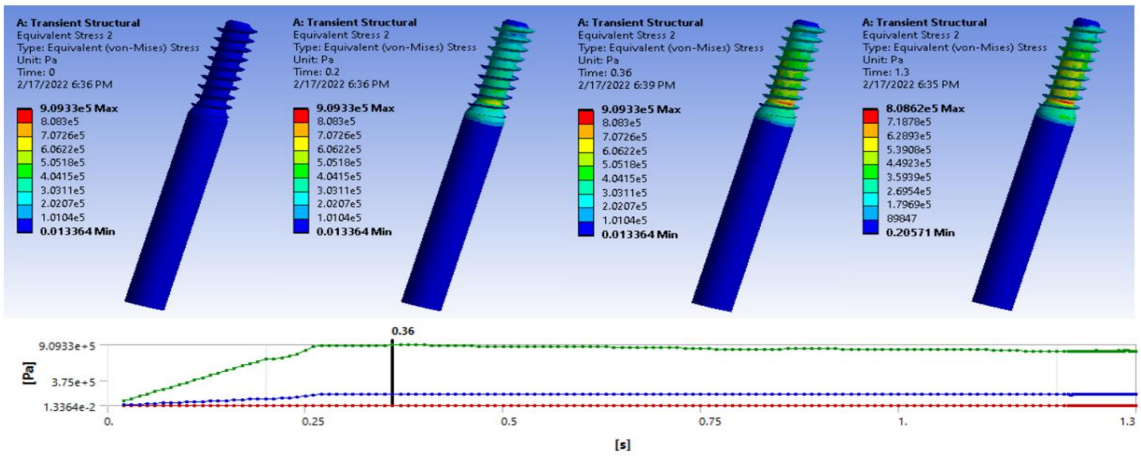


Figure 4.21:Implant Equivalent (von-Mises) Stress

During the simulation, the maximum equivalent Von-Mises stress is implemented at the base of the dynamic hip screw thread with a value of 0.9 MPa. With the tensile strength of Ti-6Al-4V being 1000 MPa, there is no remarkable reason to delve deeper into the risks of screw's failure. The points of interest of the study refer mainly to the bone strength and the contact behavior of the system.

TOTAL DEFORMATION OF THE ENTIRE MODEL

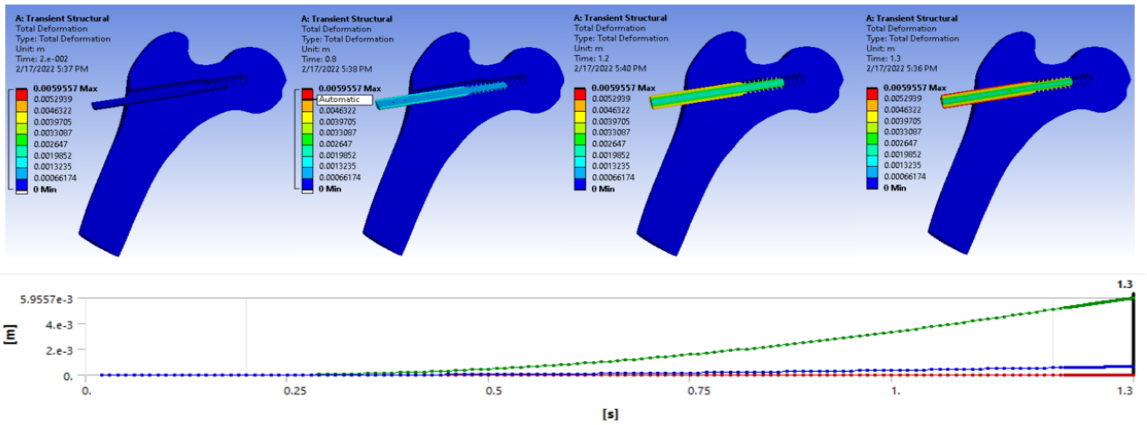


Figure 4.22:Screw Deformation

The solution of Total Deformation corresponding to all three geometries showed that the tightening of the screw had been accomplished without significant deformation of the femur parts. More specifically, the maximum deformation of the model occurred at the outer points of the dynamic hip screw with a value of almost 6 mm causing a rotation of the femoral head with the maximum deformation of it being almost 0.01mm. It is worth mentioning that a significant amount of deformation occurred during the application of the traction force showing that the deformations due to the friction forces are negligible.

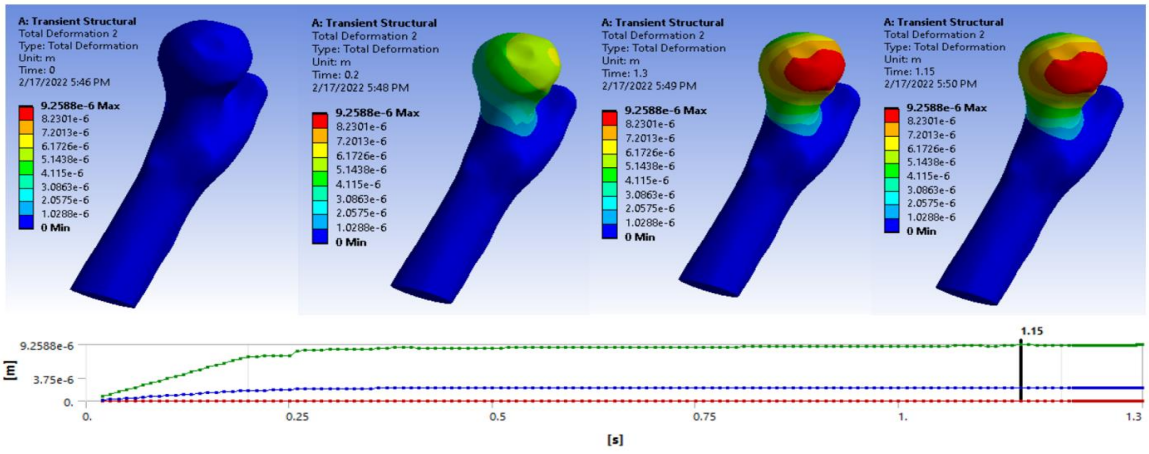


Figure 4.23:Femoral Head Deformation

MAXIMUM SLIDING DISTANCE OF THE FEMORAL HEAD SURFACE

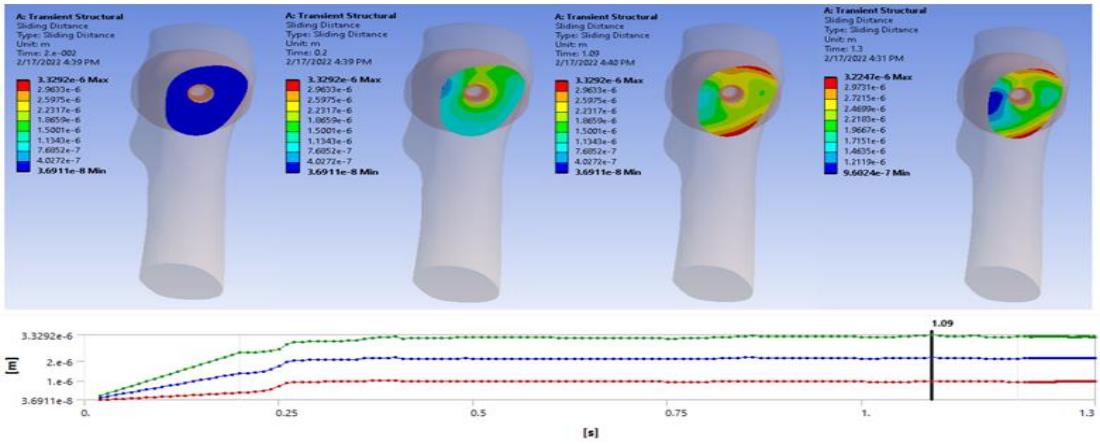


Figure 4.24:Fragment Sliding Distance

Apart from the deformation of the femoral head, conclusions can be drawn from the Sliding Distance output of the Solution. As shown above, maximum sliding distance occurred at the higher and lower points of the fracture surface with a magnitude of 0.003 mm. The magnitude of the sliding distance almost stabilizes after 0.3 sec leading to the conclusion that any further rotation of the dynamic screw didn't have any severe impact on the displacement of the femoral head. It is remarkable that the maximum slide took place at 1.09 sec.

MAXIMUM FRICTIONAL STRESS ON THE FRACTURE SURFACE

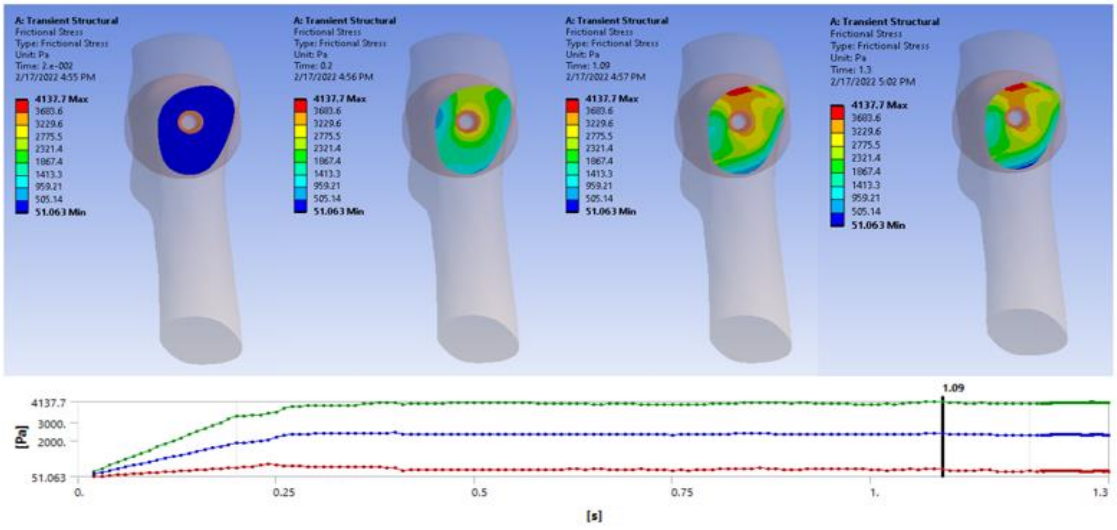


Figure 4.25:Contact Frictional Stress

The results show that maximum Frictional Stress exerted on the contact of the fragment has a value of 4.1 kPa on the higher portion of the surface, near the area of the maximum sliding distance. After the screw starts to rotate, frictional stresses are almost stabilized with small deviations.

MAXIMUM VON-MISES STRESS DISTRIBUTION OF THE FEMUR

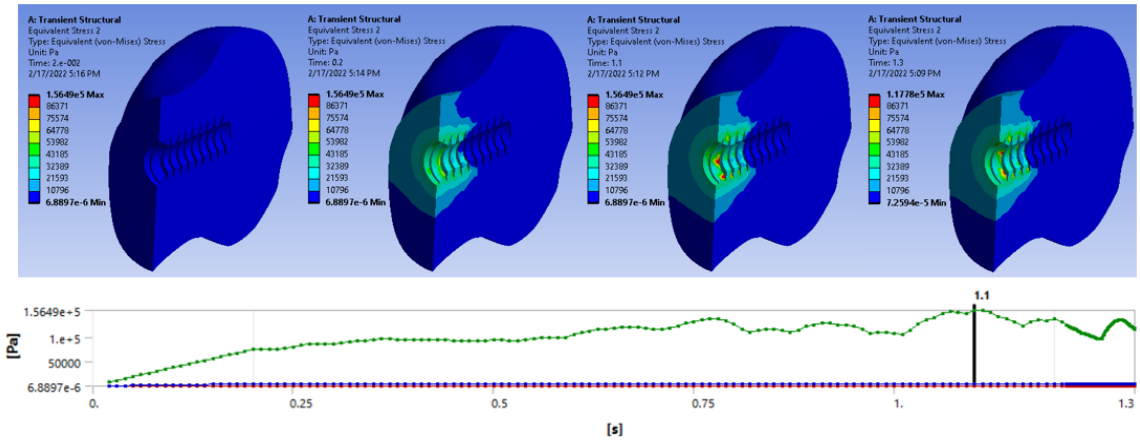


Figure 4.26:Femoral Head Equivalent (Von-Mises) Stress

Finally, Equivalent (Von-Mises) stresses had a greater value on the femoral head than on the lower femur due to stress concentration provoked by the irregular thread geometry. Maximum stresses applied inside the femoral head at the region of trabecular bone were 0.16 MPa at 1.1 sec. Considering that the trabecular bone's strength is 6.8 MPa [31], there is no restriction on the implementation of the suggested operative method.

4.5. TRANSITION TO ROBOTICS

Through the solution results, additional data considering the external forces of the screw were obtained. Data can be deployed in the subsequent simulation of the manipulator screwing the DHS, which will make the simulation more realistic. Specifically, external forces and torques applied on the screw by the bones during the insertion were calculated by the Ansys finite element analysis software and presented through the “Joint Probe” tab. Below a graph of the exerted torques on the screw is plotted. As understood, four lines are plotted to represent the moment on the X, Y, and Z-axis as red, green, and blue lines, respectively. The fourth line represents the total moment exerted on the corresponding joint, a sum of the other three.

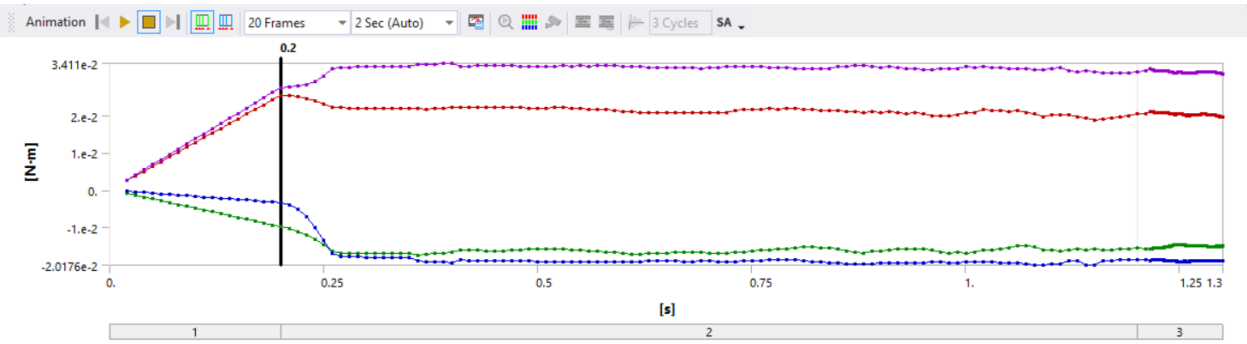


Figure 4.27: Screw's Joint Torques

To be imported at the Simulink simulation, that information should be exported from the Ansys in an Excel file (.xlsx format). The export is actualized through the tabular data, which contains the simulation results of each substep. The substeps were 205 in total, so in the following figure, only some results are shown.

Tabular Data

	Time [s]	Joint Probe (Total Moment X) [N·m]	Joint Probe (Total Moment Y) [N·m]	Joint Probe (Total Moment Z) [N·m]	Joint Probe (Total Moment Total) [N·m]
1	2.e-002	2.5406e-003	-9.1447e-004	-3.3839e-004	2.7213e-003
2	3.e-002	3.8264e-003	-1.4138e-003	-5.093e-004	4.1109e-003
3	4.e-002	5.0788e-003	-1.89e-003	-6.7815e-004	5.4613e-003
4	5.e-002	6.3823e-003	-2.3865e-003	-8.4827e-004	6.8665e-003
5	6.e-002	7.6386e-003	-2.8734e-003	-1.0167e-003	8.2242e-003
6	7.e-002	8.9466e-003	-3.3807e-003	-1.1866e-003	9.6374e-003
7	8.e-002	1.0198e-002	-3.8734e-003	-1.3548e-003	1.0993e-002
8	9.e-002	1.1494e-002	-4.3849e-003	-1.5245e-003	1.2396e-002
9	1.e-001	1.2749e-002	-4.8815e-003	-1.6928e-003	1.3756e-002
10	0.11	1.4041e-002	-5.3887e-003	-1.8622e-003	1.5155e-002
11	0.12	1.5304e-002	-5.8824e-003	-2.0303e-003	1.652e-002
12	0.13	1.6592e-002	-6.3915e-003	-2.1991e-003	1.7916e-002
13	0.14	1.7858e-002	-6.8899e-003	-2.3667e-003	1.9287e-002
14	0.15	1.9144e-002	-7.3978e-003	-2.5348e-003	2.068e-002
15	0.16	2.0417e-002	-7.9015e-003	-2.7019e-003	2.2059e-002
16	0.17	2.1699e-002	-8.4089e-003	-2.8691e-003	2.3448e-002
17	0.18	2.2972e-002	-8.911e-003	-3.0353e-003	2.4826e-002
18	0.19	2.4249e-002	-9.4138e-003	-3.2014e-003	2.6208e-002
19	0.2	2.5523e-002	-9.9125e-003	-3.3663e-003	2.7586e-002
20	0.21	2.5487e-002	-1.0452e-002	-4.0273e-003	2.784e-002
21	0.22	2.5161e-002	-1.1118e-002	-5.232e-003	2.8001e-002
22	0.23	2.4663e-002	-1.2056e-002	-7.2404e-003	2.8391e-002
23	0.24	2.3951e-002	-1.3235e-002	-1.0013e-002	2.9138e-002
24	0.25	2.311e-002	-1.4686e-002	-1.3536e-002	3.0544e-002
25	0.26	2.2398e-002	-1.6396e-002	-1.7037e-002	3.2569e-002
26	0.27	2.2146e-002	-1.6906e-002	-1.7781e-002	3.3052e-002
27	0.28	2.2065e-002	-1.6967e-002	-1.7886e-002	3.3085e-002
28	0.29	2.1993e-002	-1.7062e-002	-1.7981e-002	3.3138e-002
29	0.3	2.192e-002	-1.7095e-002	-1.8052e-002	3.3145e-002
30	0.31	2.1914e-002	-1.7093e-002	-1.8104e-002	3.3168e-002
31	0.32	2.1928e-002	-1.7071e-002	-1.8149e-002	3.3191e-002
32	0.33	2.1953e-002	-1.706e-002	-1.8197e-002	3.3228e-002

Graph Tabular Data

Figure 4.28: Tabular Data of Moments

The method by which those values are utilized for the manipulator's control system will be described in the next chapter.

5. DESIGN OF DYNAMIC HIP SCREW INSERTION USING ROBOTIC ARM

5.1. INTRODUCTION

Concerning the second part of the thesis, a hybrid position/force control has been designed to execute the process of the DHS insertion simulated in Ansys. This process consists of the exertion of a tractive force on the screw through the force controller and the initiation of the rotation of the screw until it obtains a steady angular velocity through the position controller. The reaction forces of the screw from the bones, calculated by the Ansys, were imported at the last link of the Kuka through the SimScape Multibody toolbox as external forces. In this chapter, some basic kinematic equations of the Kuka IIWA will be calculated, and then all the procedures for creating the Kuka IIWA in the Mechanical Explorer and designing its control system will be analyzed.

5.2. SIMULINK MODEL

5.2.1. KUKA DENAVIT-HARTENBERG EQUATIONS

Denavit-Hartenberg introduced 1955 an algorithm for forming kinematic equations for robotic N-arms of degrees of freedom. According to him, we initially assume that a robotic arm with N degrees of freedom has N links connected by N joints [32].

First, attachment of the general stationary coordinate system and the local coordinate systems of each joint should be implemented with the method described below:

- The Z_i axis coincides with the axis of the joint i, while the direction of the Z_i axis is set arbitrarily.

- The Xi axis is defined by the common perpendicular between Z_i and Z_{i+1} and the direction from Z_i to Z_{i+1}.
- The origin O_i of the coordinate system {i} is defined by the intersection of X_i and Z_i.
- The outer product of Zi defines the Yi axis on Xi since we have an orthogonal right-handed Cartesian coordinate system.
- The coordinate system of the end-effector (last link) is defined by the X_N and Z_N axis. The point of intersection and orientation of the X_N are arbitrary but are usually defined in such a way as to facilitate the definition of the work performed by the robot.

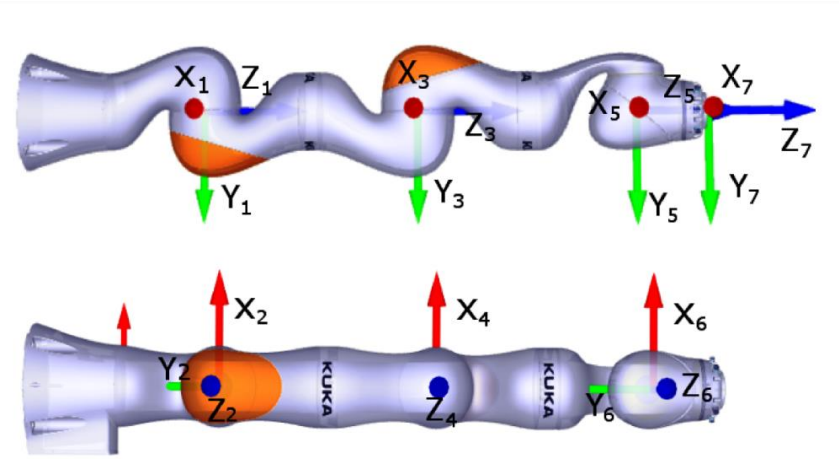


Figure 5.1: Coordinate Systems [33]

After the attachment of the coordinate systems to each joint has been completed, the values of the four kinematic characteristics of the joints can be easily extracted. Each link has two fixed features and two features that describe how each link connects to its neighbors. These features are:

Stable features:

1) **Link length (L_i):** the length of the line segment that intersects the Z_i and Z_{i+1} axes of the joints at the points O_i and O_{i+1}, respectively.

2) **Link twist (α_i):** the angle between the Z_i to the Z_{i+1} axes of the two joints around the X_i axes.

Connection features:

1) **Link offset (d_i):** is measured on the Z_i axis of the joint and is the distance of the line segment that intersects the axis X_i and X_{i-1}.

2) **Joint angle (θ_i):** the angle of rotation of the X_i to X_{i-1} axes that is measured on the Z_i axis

These characteristics above are sufficient to describe kinematically each robotic arm. Three of those quantities are constant for a given link, while the fourth parameter, θ_i for a revolute joint or d_i for a prismatic joint, is the joint variable.

$$\text{Rotary joint} \begin{cases} \text{Joint variable} & q_1 = \theta_i \\ \text{Link variables} & \alpha_i \;\; L_i \;\; d_i \end{cases}$$

$$\text{Translational joint} \begin{cases} \text{Joint variable} & q_1 = d_i \\ \text{Link variables} & \alpha_i \;\; L_i \;\; \theta_i \end{cases}$$

Link *i*	**θ *(°)***	**d *(m)***	**a *(m)***	**α *(°)***
1	θ_1	0.34	0	-90
2	θ_2	0	0	90
3	θ_3	0.4	0	-90
4	θ_4	0	0	90
5	θ_5	0.4	0	-90
6	θ_6	0	0	90
7	θ_7	0.125	0	0

Table 5.1: Kuka D-H Parameters [34]

Since, from one local coordinate system to the next, there are two kinds of displacements (L_i, d_i) and two kinds of rotations (α_i, θ_i), the total transformation of one system into another will be the

product of these four corresponding homogeneous transformations. So, after recording all the four parameters of each link, we proceed to a simple replacement in the $^{i-1}M_i$ transformation table:

$$ {}^{i-1}M_i = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & \alpha_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & \alpha_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 5.1 $$

After replacing the D-H parameters of the Kuka IIWA manipulator presented above, into the transformation table, for each link, the following homogeneous matrices result.

```
M1 =

[cos(th1),  0, -sin(th1),  0]
[sin(th1),  0,  cos(th1),  0]
[       0, -1,         0, d1]
[       0,  0,         0,  1]

M2 =

[cos(th2), 0,  sin(th2), 0]
[sin(th2), 0, -cos(th2), 0]
[       0, 1,         0, 0]
[       0, 0,         0, 1]

M3 =

[cos(th3),  0, -sin(th3),  0]
[sin(th3),  0,  cos(th3),  0]
[       0, -1,         0, d3]
[       0,  0,         0,  1]

M4 =

[cos(th4), 0,  sin(th4), 0]
[sin(th4), 0, -cos(th4), 0]
[       0, 1,         0, 0]
[       0, 0,         0, 1]

M5 =

[cos(th5),  0, -sin(th5),  0]
[sin(th5),  0,  cos(th5),  0]
[       0, -1,         0, d5]
[       0,  0,         0,  1]

M6 =

[cos(th6), 0,  sin(th6), 0]
[sin(th6), 0, -cos(th6), 0]
[       0, 1,         0, 0]
[       0, 0,         0, 1]

M7 =

[cos(th7), -sin(th7), 0,  0]
[sin(th7),  cos(th7), 0,  0]
[       0,         0, 1, d7]
[       0,         0, 0,  1]
```

Figure 5.2: Homogeneous Matrices of Kuka's Links

After completing the transformation tables for all the local coordinate systems, we proceed to the calculation of their product $^0M_N = {}^0M_1\ {}^1M_2\ {}^2M_3 \ldots {}^{N-1}M_N$ and finally to the creation of the total ^0M_N transformation table. From the resulting (4x4) table, the upper left (3x3) table and the fourth column symbolize the orientation and position of the end-effector, respectively, in relation to the stationary system of the base.

After multiplying the above matrices, the total homogeneous transformation is obtained, through Matlab, as follows:

$$ {}^{0}M_7 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad 5.2 $$

Where:

$\mathbf{r_{11}} = c_7(s_6(s_4(s_1s_3 - c_1c_2c_3) - c_1c_4s_2) - c_6(c_5(c_4(s_1s_3 - c_1c_2c_3) + c_1s_2s_4) + s_5(c_3s_1 + c_1c_2s_3))) + s_7(s_5(c_4(s_1s_3 - c_1c_2c_3) + c_1s_2s_4) - c_5(c_3s_1 + c_1c_2s_3))$,

$\mathbf{r_{12}} = c_7(s_5(c_4(s_1s_3 - c_1c_2c_3) + c_1s_2s_4) - c_5(c_3s_1 + c_1c_2s_3)) - s_7(s_6(s_4(s_1s_3 - c_1c_2c_3) - c_1c_4s_2) - c_6(c_5(c_4(s_1s_3 - c_1c_2c_3) + c_1s_2s_4) + s_5(c_3s_1 + c_1c_2s_3)))$,

$\mathbf{r_{13}} = - c_6(s_4(s_1s_3 - c_1c_2c_3) - c_1c_4s_2) - s_6(c_5(c_4(s_1s_3 - c_1c_2c_3) + c_1s_2s_4) + s_5(c_3s_1 + c_1c_2s_3))$,

$\mathbf{r_{21}} = - c_7(s_6(s_4(c_1s_3 + c_2c_3s_1) + c_4s_1s_2) - c_6(c_5(c_4(c_1s_3 + c_2c_3s_1) - s_1s_2s_4) + s_5(c_1c_3 - c_2s_1s_3))) - s_7(s_5(c_4(c_1s_3 + c_2c_3s_1) - s_1s_2s_4) - c_5(c_1c_3 - c_2s_1s_3))$,

$\mathbf{r_{22}} = s_7(s_6(s_4(c_1s_3 + c_2c_3s_1) + c_4s_1s_2) - c_6(c_5(c_4(c_1s_3 + c_2c_3s_1) - s_1s_2s_4) + s_5(c_1c_3 - c_2s_1s_3))) - c_7(s_5(c_4(c_1s_3 + c_2c_3s_1) - s_1s_2s_4) - c_5(c_1c_3 - c_2s_1s_3))$,

$\mathbf{r_{23}} = c_6(s_4(c_1s_3 + c_2c_3s_1) + c_4s_1s_2) + s_6(c_5(c_4(c_1s_3 + c_2c_3s_1) - s_1s_2s_4) + s_5(c_1c_3 - c_2s_1s_3))$,

$\mathbf{r_{31}} = s_7(s_5(c_2s_4 + c_3c_4s_2) + c_5s_2s_3) - c_7(c_6(c_5(c_2s_4 + c_3c_4s_2) - s_2s_3s_5) + s_6(c_2c_4 - c_3s_2s_4))$,

$\mathbf{r_{32}} = c_7(s_5(c_2s_4 + c_3c_4s_2) + c_5s_2s_3) + s_7(c_6(c_5(c_2s_4 + c_3c_4s_2) - s_2s_3s_5) + s_6(c_2c_4 - c_3s_2s_4))$,

$\mathbf{r_{33}} = c_6(c_2c_4 - c_3s_2s_4) - s_6(c_5(c_2s_4 + c_3c_4s_2) - s_2s_3s_5)$,

$\mathbf{t_x} = d_3c_1s_2 - d_5(s_4(s_1s_3 - c_1c_2c_3) - c_1c_4s_2) - d_7(c_6(s_4(s_1s_3 - c_1c_2c_3) - c_1c_4s_2) + s_6(c_5(c_4(s_1s_3 - c_1c_2c_3) + c_1s_2s_4) + s_5(c_3s_1 + c_1c_2s_3)))$,

$\mathbf{t_y} = d_7(c_6(s_4(c_1s_3 + c_2c_3s_1) + c_4s_1s_2) + s_6(c_5(c_4(c_1s_3 + c_2c_3s_1) - s_1s_2s_4) + s_5(c_1c_3 - c_2s_1s_3))) + d_5(s_4(c_1s_3 + c_2c_3s_1) + c_4s_1s_2) + d_3s_1s_2$,

$\mathbf{t_z} = d_1 + d_5(c_2c_4 - c_3s_2s_4) - d_7(s_6(c_5(c_2s_4 + c_3c_4s_2) - s_2s_3s_5) - c_6(c_2c_4 - c_3s_2s_4)) + d_3c_2$.

Those equations are used from the Forward Kinematic subsystem, which will be analyzed in the next chapter. The inputs of the block are the sensed joint angles while the outputs are the t_x, t_y, t_z which concerns the position of the end-effector each time, and the Euler angles as a function of $\mathbf{r_{11}}$, $\mathbf{r_{12}}$, $\mathbf{r_{13}}$, $\mathbf{r_{21}}$, $\mathbf{r_{22}}$, $\mathbf{r_{23}}$, $\mathbf{r_{31}}$, $\mathbf{r_{32}}$, and $\mathbf{r_{33}}$.

5.2.2. KUKA MODEL

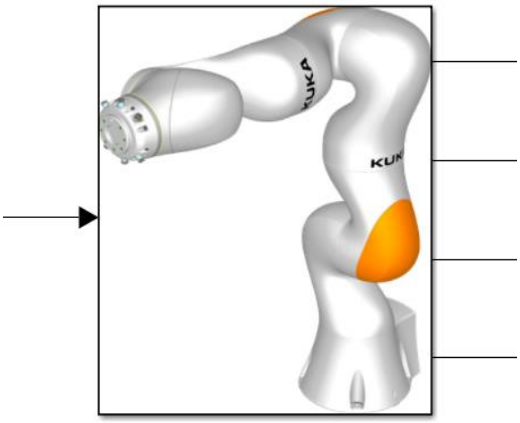


Figure 5.3: Kuka IIWA Subsystem

Kuka IIWA is a robotic arm that can be simulated with a multi-physics model. In the present thesis, the properties that correspond to the area of interest are the kinematic and dynamic properties. Based on those properties, the Inverse Dynamics, Kinematics, and Jacobian Matrices will be calculated so that the Kuka will perform the desired motions. The import of the Kuka IIWA robot into the Simulink, was accomplished through the Simscape Multibody toolbox. The:

robot=importrobot('iiwa7.urdf');

line was initially used so that to load the following blocks to Simulink.

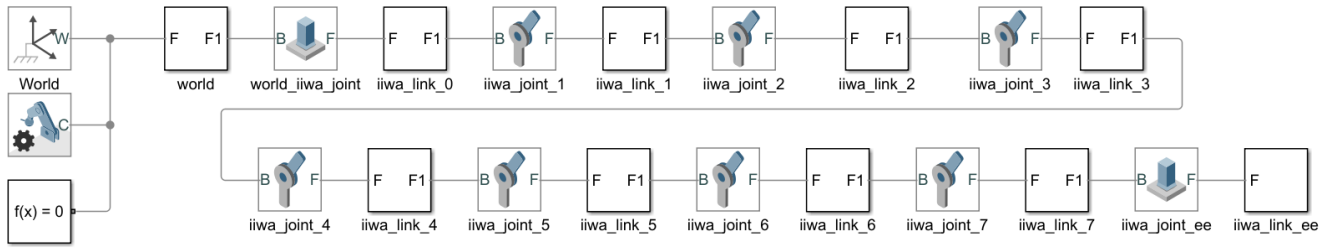


Figure 5.4:Kuka Manipulator

The first three blocks represent from top to bottom, the “World Frame”, which gives access to the unique and motionless world frame, “Mechanism Configuration”, which sets global mechanical parameters like gravity, and “Solver Configuration”, which gives general solver setting for the simulation.

Those three blocks are then connected linearly with a sequence of joints and blocks. Specifically, the first joint is a “Weld Joint”, indicating the base of the Kuka, while there are seven more “Revolute Joints” representing the seven degrees of freedom of the manipulator. Those joints relate to each other via blocks that introduce the transport and angular transformation of coordinate systems between two following joint’s coordinate systems. Those transformations are accomplished through “Rigid Transform” blocks, as shown in the first line of the following image. Apart from the origin and axis transform of each joint, those blocks also introduce the visualization and the dynamic characteristics of the links, as shown in the two successive lines of the image. Specifically, the second line introduce the transformation for the .stl file to fit at the appropriate point, while the third line introduce the transformation for the center of mass.

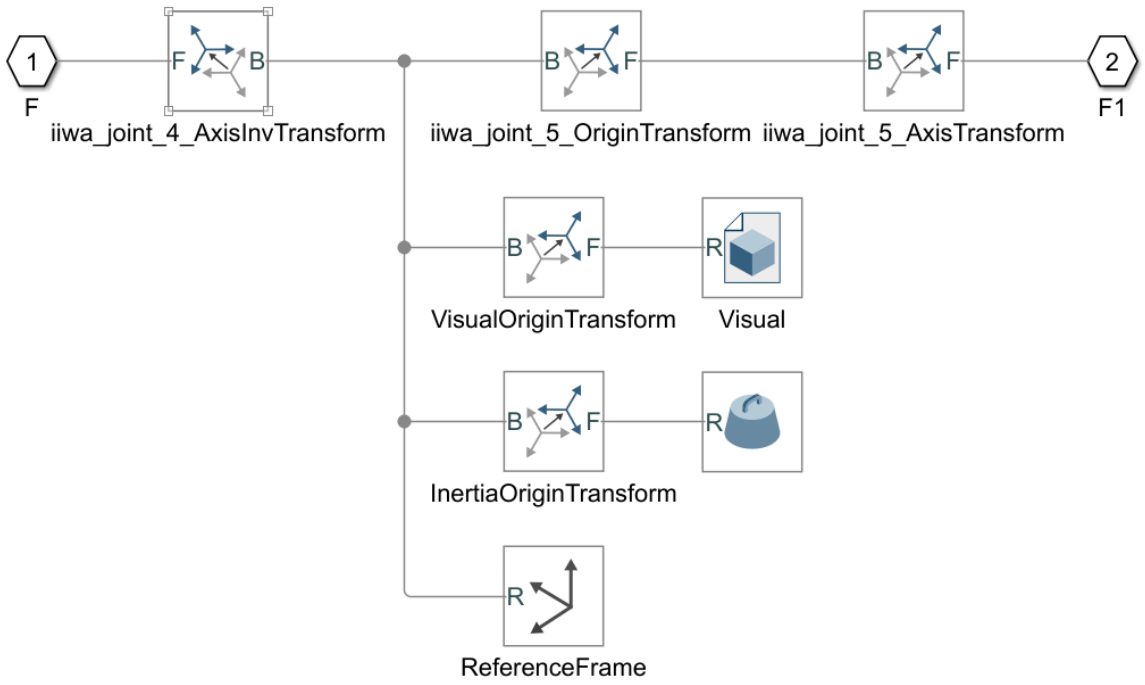


Figure 5.5: iiwa_link_4 block

This thesis is only devoted to implementing a control system for the Kuka to be used for a surgical operation. No emphasis has been given to designing or exploiting an end-effector tool. Considering revolute joints, every block has internal mechanics (Spring and Damping Coefficient) and limit bound parameters specifying the behavior of the joint. So, for the Kuka end-effector to perform multi-rotational moves, the angle bounds of the seventh rotational joint have been designated to have a range from -100 to +100 rad has been defined, as shown below.

The actuation method has been specified to be based on the torque input. While sensing capabilities have been activated to use the values of joint configurations, velocities, accelerations and torques for the closed-loop controller.

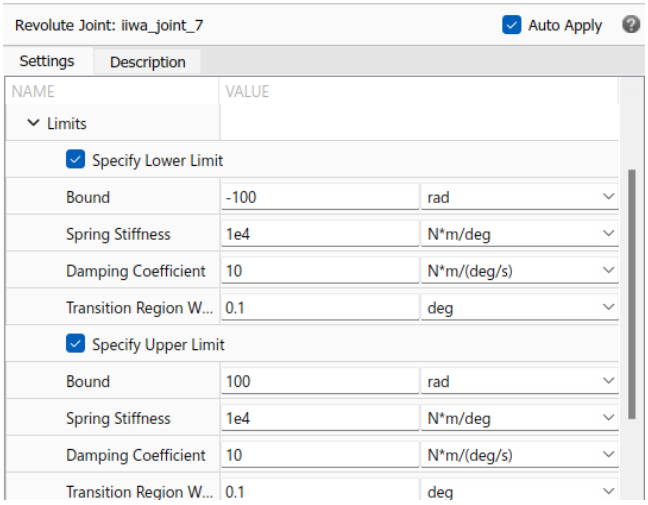
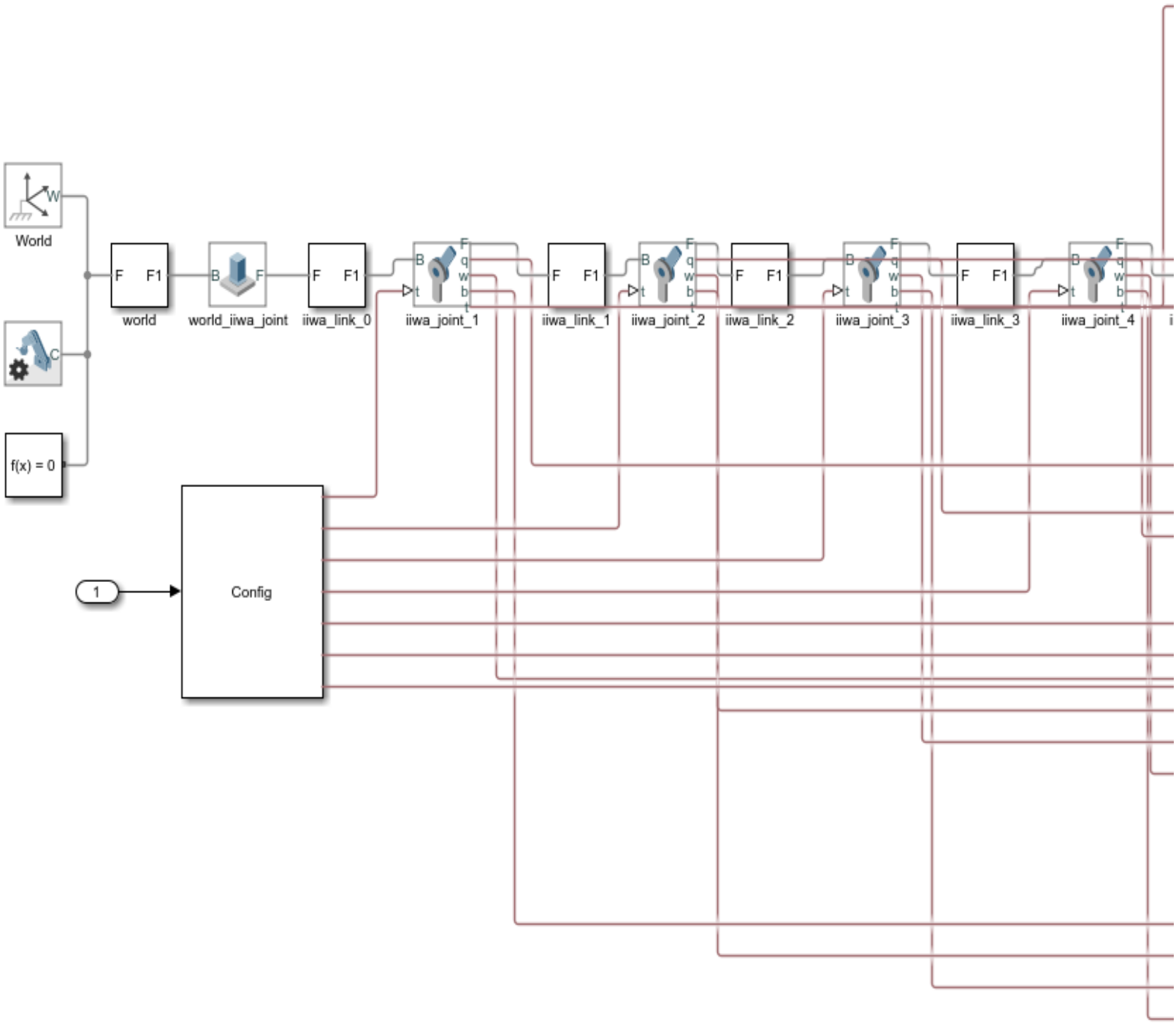
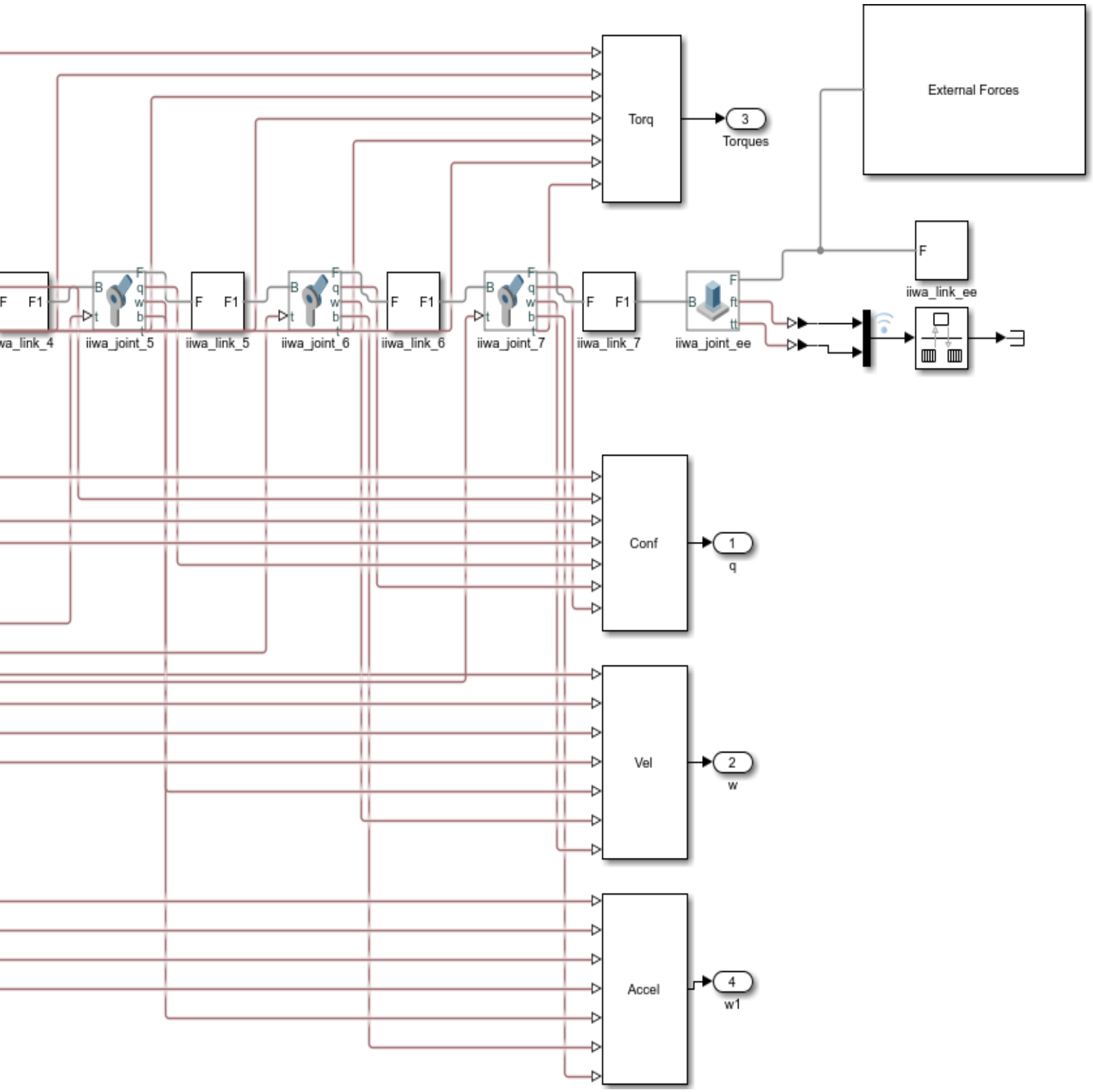


Figure 5.6: Seventh Joint Limits

The total diagram of the Kuka, including external forces, activation, and sensors wires designed through SimScape Multibody, is presented below.


World
world
world_iiwa_joint
iiwa_link_0
iiwa_joint_1
iiwa_link_1
iiwa_joint_2
iiwa_link_2
iiwa_joint_3
iiwa_link_3
iiwa_joint_4
f(x) = 0
1
Config


Torq
3
Torques
External Forces
F
iiwa_link_ee
iiwa_link_4
iiwa_joint_5
iiwa_link_5
iiwa_joint_6
iiwa_link_6
iiwa_joint_7
iiwa_link_7
iiwa_joint_ee
Conf
1
q
Vel
2
w
Accel
4
w1

Finally, the external forces exerted on the end-effector during the operation are generated into the "External Forces" subsystem and applied at the end-effector.



Figure 5.7: External Forces Subsystem

Inside the "External Forces" subsystem, a block diagram has been designed to implement the forces and torques acquired from the finite element analysis at the end-effector.

The "Signal Builder" blocks have been used in the subsystem, in which the forces' and torques' Excel files are uploaded, and based on the time displayed on the first column and the values next of those times, the respective signals are generated.

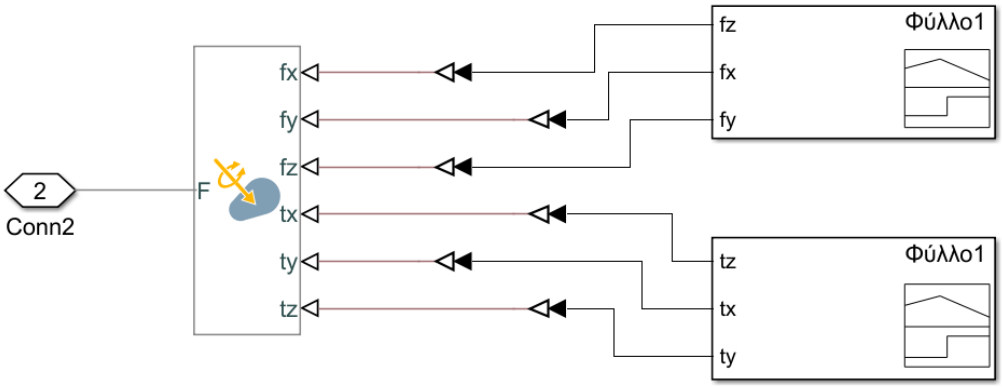


Figure 5.8: External Forces Diagram

With a quick comparison with the diagrams in chapter 4.3., it is easily understood that the signals are the same. An internal change has only taken place so that the different axes fit. For instance, the z-axis of the Ansys' results corresponds to the x-axis of the Simulink simulation. The Signals generated in the Moment's "Signal Builder" are presented below.

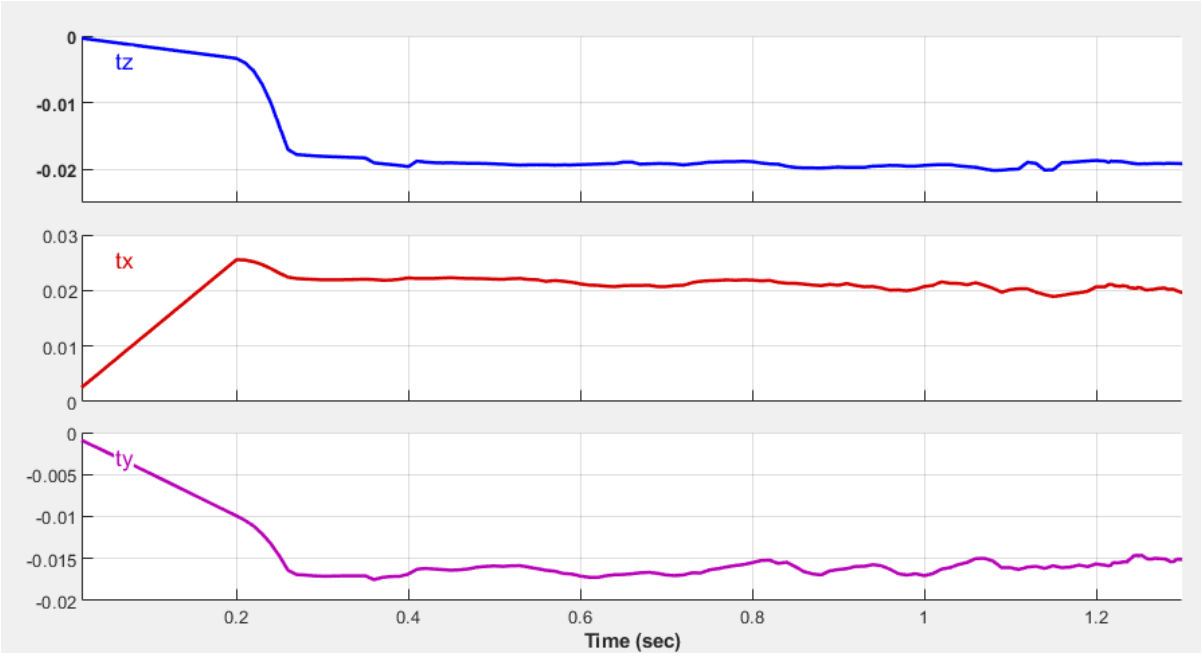


Figure 5.9: Moments in Building Signals

5.2.3. POSITION CONTROL



Figure 5.10: Position Control Subsystem

Joint-Space control satisfies only the free from external forces motion of the manipulator. The task-space control should be implemented if the manipulator interacts with the environment. The end-effector's position should be calculated in each step to compare it with the desired one. In this

case, joint configurations are used to apply forward kinematics to find position and orientation. However, only position and the Euler angles can be calculated through forward kinematics. For this reason, Luh, Walker, and Paul [35] suggested a new method of calculating the error of orientation of the end effector. The equation is shown below:

$$\underline{\tilde{\boldsymbol{\varphi}}} = \frac{1}{2}\left[{}^{u}\underline{\mathbf{i}}\,(t) \times {}^{u}\underline{\mathbf{i}}_{d}(t) + {}^{u}\underline{\mathbf{j}}\,(t) \times {}^{u}\underline{\mathbf{j}}_{d}(t) + {}^{u}\underline{\mathbf{k}}(t) \times {}^{u}\underline{\mathbf{k}}_{d}(t) \right] \qquad 5.3$$

Where ${}^{u}\underline{\mathbf{i}}$, ${}^{u}\underline{\mathbf{j}}$, ${}^{u}\underline{\mathbf{k}}$ are the unit vectors of the rotation matrix **R** of the end-effector and ${}^{u}\underline{\mathbf{i}}_{d}$, ${}^{u}\underline{\mathbf{j}}_{d}$, ${}^{u}\underline{\mathbf{k}}_{d}$, the equivalent desired values.

In Simulink, the orientation error of the end-effector is calculated through this equation. The subsystem in which the rotation error was calculated is called “Rotation False’’. In this subsystem, the two vectors of the Desired Angles and the Real Angles are imported, while the Rotation Error is exported. For the accumulation of the six desired vectors used in the equation above, two “Coordinate Transformation Conversion” blocks are used. One for converting the current actual Euler Angles of the manipulator into Rotational Matrix, and one for converting the desired Euler Angles into Rotational Matrix. Those two matrices were used as input to a “Matlab Function” block in which the above equation was composed.

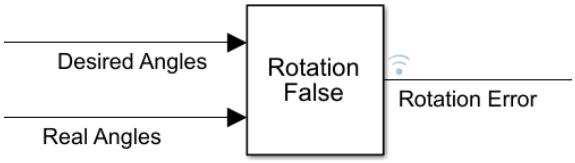


Figure 5.11: Rotation False Subsystem

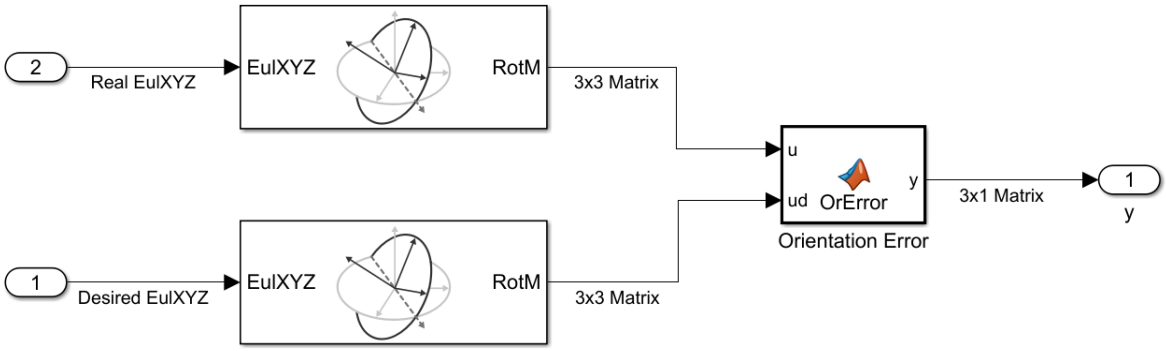


Figure 5.12: Rotation False Diagram

The equation written in "Matlab Function" is shown in the following lines:

```
function y = OrError( u , ud )
i=u(1:3,1);
j=u(1:3,2);
k=u(1:3,3);
id=ud(:,1);
jd=ud(:,2);
kd=ud(:,3);
f=1/2*[cross(i,id)+cross(j, jd)+cross(k, kd)];
y=f;
```

Though, the total vector of the error between the desired and the real position and orientation is given by the equation:

$$\tilde{\mathbf{r}} = \begin{bmatrix} \underline{\tilde{\boldsymbol{p}}} \\ \tilde{\boldsymbol{\varphi}} \end{bmatrix} \qquad 5.4$$

Where $\underline{\tilde{\mathbf{p}}}(t) = \underline{\mathbf{p}}_d(t) - \underline{\mathbf{p}}(t)$ is the position error.

The position error comes right after the deduction of the actual position (as calculated from the "Forward Kinematics") from the desired position defined by the user. The desired position signal is created by the "Signal Builder" block. The signal is the same as at the Ansys simulation as shown below. In the following diagram, the total Position Control System is displayed. The current actual position of the end-effector is imported in a 6x1 vector and divided into two 3x1 orientation and position vectors. The first is imported into the "Rotation False" subsystem studied above, while the other is straight deducted from the desired position. The resulting errors enter the PID controllers, which export the required moments and forces needed to be applied at the end-effector by the manipulator. Those two signals are merged through the "Matrix Concatenate" block and multiplied with the transposed Geometric Jacobian matrix created with the "Get Jacobian" block based on the current joint positions. The proper joint torques are acquired through this multiplication to accomplish the desired forces and moments at the end-effector. So, a vector with dimension of 7x1 is the output of the position controller.

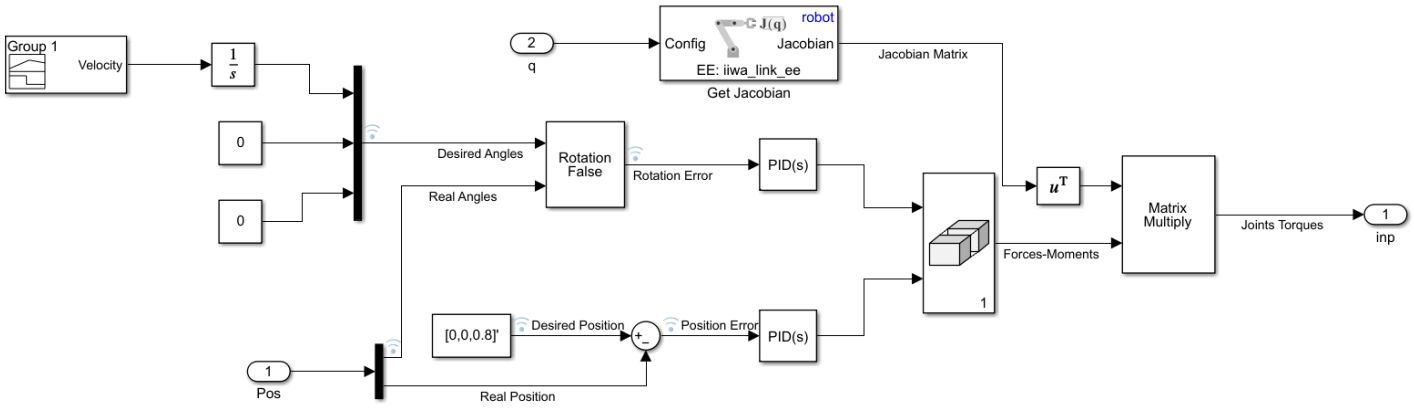


Figure 5.13: Position Control Block Diagram

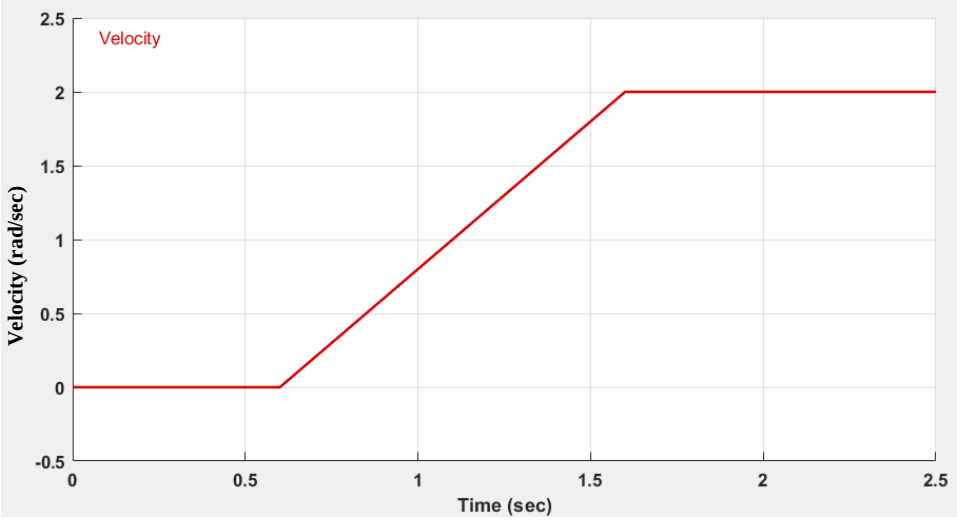


Figure 5.14: Rotational Velocity

5.2.4. DIRECT FORCE CONTROL



Figure 5.15: Force Control Subsystem

The implementation of the force control will be based on the equation occurred by the direct force control of one degree of freedom. Let us assume that there is a system of a mass m, which exerts a f_e force on an environment. The mass, in our case, is the end-effectors tool, namely the screw. (In this thesis, any screw's gripper is not designed, so we consider that screw is attached right at the end of the seventh link of Kuka). Considering f_u the desired force exerted on the mass from the force control system, we take the equation[36]:

$$m\ddot{x} = f_u - f_e + f_d \qquad 5.5$$

External force f_e is calculated by a sensor at manipulator's the end-effector. We will use a simple mass-spring model; thus, the external force can be a function of the k_e environment's coefficient of elasticity. In our case, the environment is the bone, and since the screw interacts with the

cancellous bone, we consider the k_e as the cancellous' bone elastic modulus used in the previous chapter.

$$f_e = k_e \tilde{x}_e \tag{5.6}$$

Symbolizing the error between the derided and the real force with $\tilde{f}$ we have:

$$\tilde{f} = f_d - f_e$$
$$\ddot{\tilde{f}} = \ddot{f}_d - \ddot{f}_e = \ddot{f}_d - k_e \ddot{\tilde{x}}_e = \ddot{f}_d - k_e (f_u - f_e + d)/m \tag{5.7}$$

Choosing a control law that the system's response will have a second order response to the $\tilde{f}$ error.

$$\ddot{\tilde{f}} + k_d \dot{\tilde{f}} + k_p \tilde{f} + k_i \int \tilde{f} = -k_e d/m \tag{5.8}$$

Replacing in the last equation the one before and writing it in terms of the variables that we are interested we arrive at the force control.

$$f_u = f_e + \frac{m}{k_e}\left(\ddot{f}_d + k_d \dot{\tilde{f}} + k_p \tilde{f} + k_i \int \tilde{f}\right) \tag{5.9}$$

The above equation is formed in the Simulink, as shown in the following figure. On the left, the desired forces as defined by the user. Expressly, the first three values representing the moments exerted by the end-effector, and the last two values representing the forces on the y- and z-axis, are zeros. The fourth variable, indicating the force on the x-axis, is inserted through the "Signal Builder" block. The created signal is the same as the one in Ansys and it is plotted below. The vector of the six real moments and forces are deducted from the desired ones through the "Sum" block, and the resulting 6 component vector passes through the "PID controller". The output of this block is multiplied by the second derivative of the desired forces. Finally, the product is multiplied by a constant representing the mass of the tool on the end-effector divided by the elastic modulus of the environment, as defined in the "m/k" block. The output of the "Matrix Multiply" block is finally summed with the real forces as calculated by the sensor. The output of the "Sum" block is the vector of actual forces that the controller commands the end-effector to exert. Of

course, those forces should be translated into joint torques, so they are multiplied with the transpose of the Geometric Jacobian matrix, as it raised from the “Get Jacobian” block based on the current joint configuration.

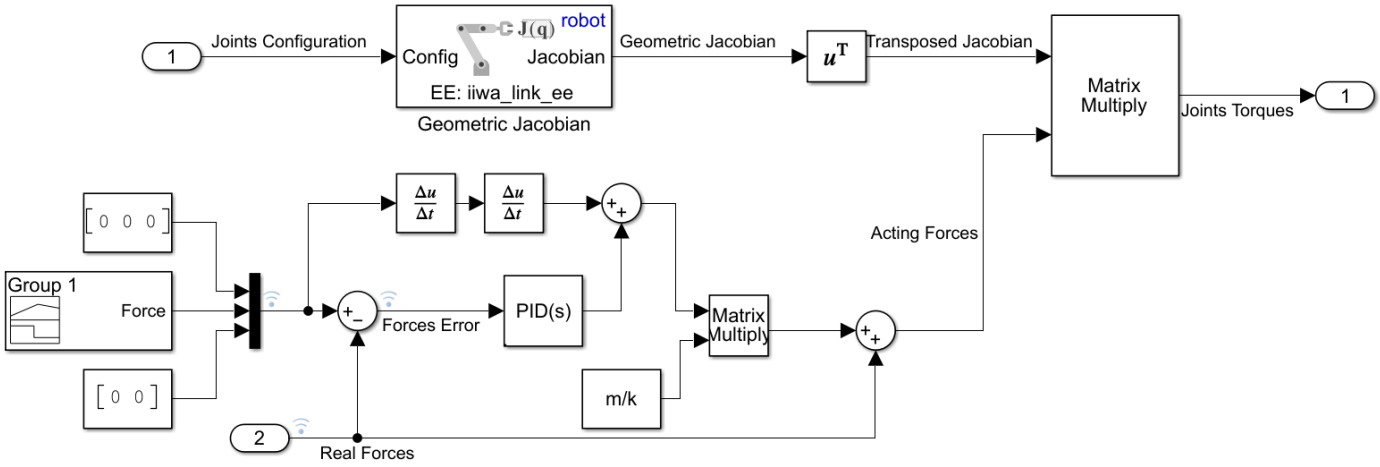


Figure 5.16: Direct Force Control Diagram

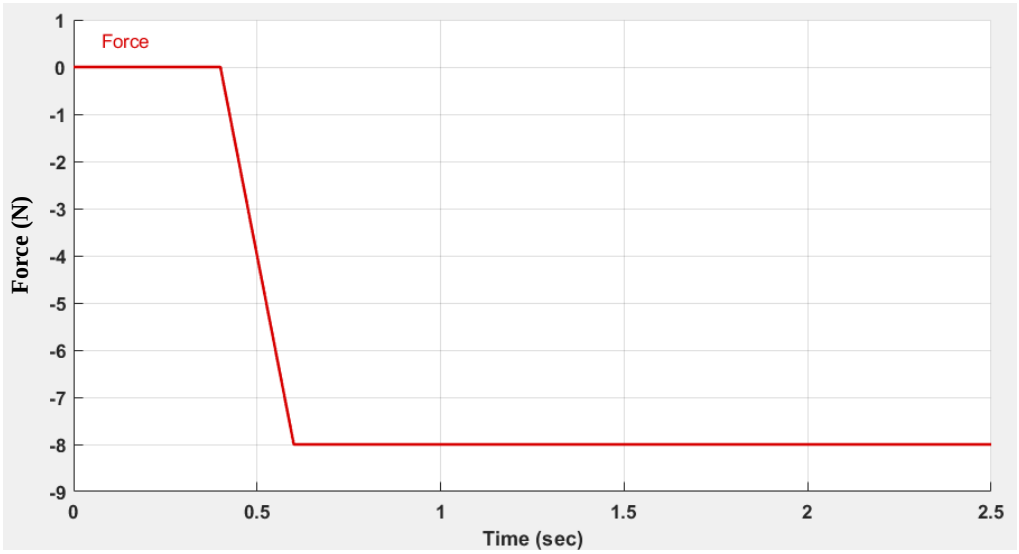


Figure 5.17: Desired Force Signal

5.2.5. HYBRID POSITION/FORCE CONTROL

When the application of the manipulator concerns a different type of controlling (position/force) on the six degrees of freedom, then there is a method of differentiating the control. In applying the dynamic hip screw insertion, described above, we should control the x-axis corresponding to the desired force. In contrast, the rotation of the screw around the x-axis should be accomplished through position control. The other four degrees of freedom will also be controlled through the position control, wanting the two orientations and positions to be constant throughout the simulation.

To achieve this separation of the degrees of freedom between the position and the force control, the two 6x6 diagonal matrices are used with ones and zeros in the diagonal. Specifically, where one is present in the one matrix, a zero is present in the other, indicating that each degree of freedom can be controlled only by one controller.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 5.18: Position's Control Matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 5.19: Force's Control Matrix

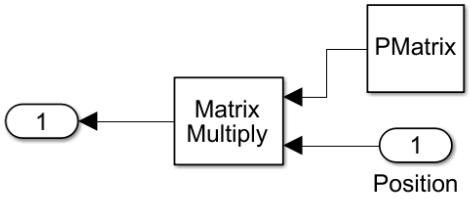


Figure 5.20: 5 DoF ← 6 DoF

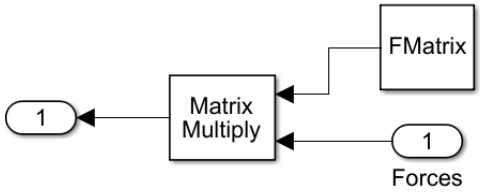


Figure 5.21: 1 DoF ← 6 DoF

The matrices shown above indiate that the three orientational and the two positional degrees of freedom will pass through the position controller while only one through the force controller.

In the completed control system presented below, the two Subsystems called 5 DoF ← 6 DoF and 1 DoF ← 6 DoF represent the product of the position and the forces with the corresponding matrices.

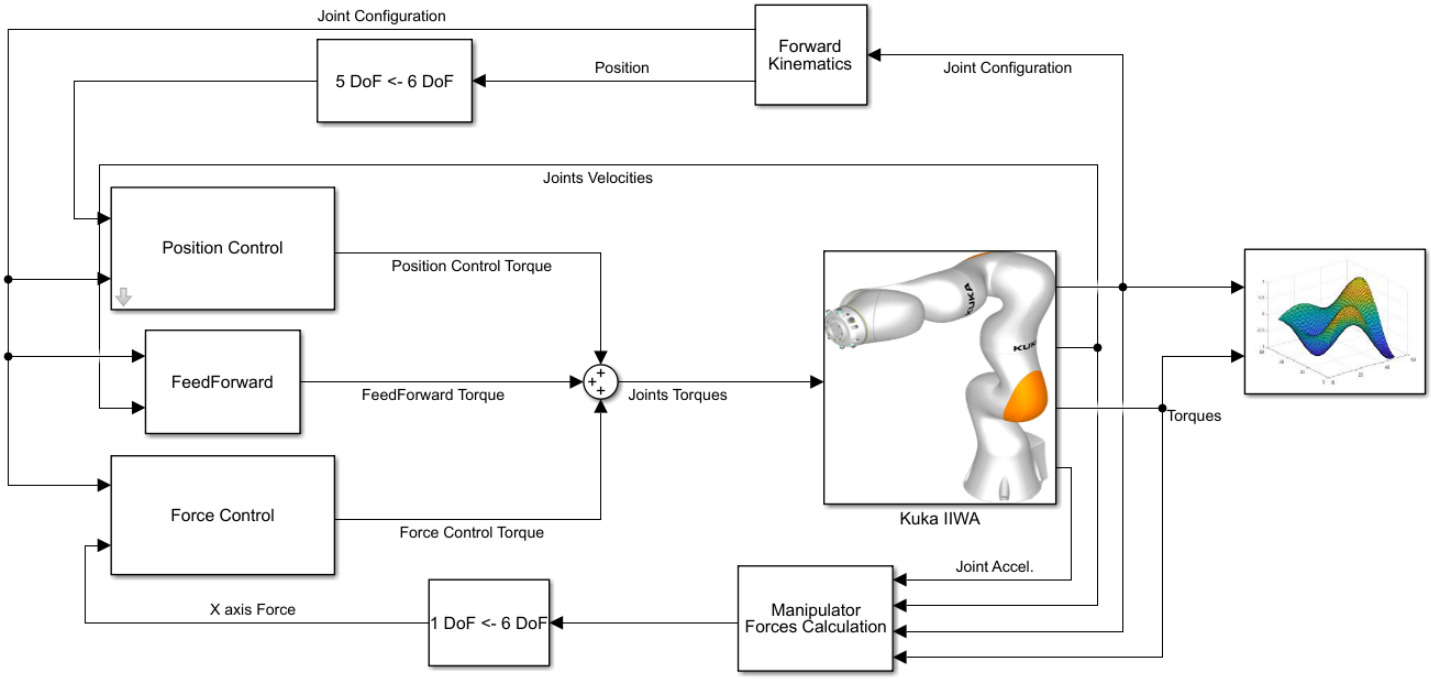


Figure 5.22: System Diagram

So, only one degree of freedom is controlled by the Force Control subsystem, while five by the Position Control subsystem.

5.2.6. FEEDFORWARD CONTROL



Figure 5.23: Feedforward Subsystem

As shown in the system diagram, other than the torques from the hybrid control, the manipulators receive feedforward torques calculated in the "FeedForward" subsystem. This subsystem concludes the "Gravity Torque" and "Velocity Product Torque" blocks. The first one calculates the required joint torques to compensate for gravity for the given configuration. In contrast, the second one calculates joint torques that cancel velocity-induced forces, like Coriolis, centrifugal and frictional forces.

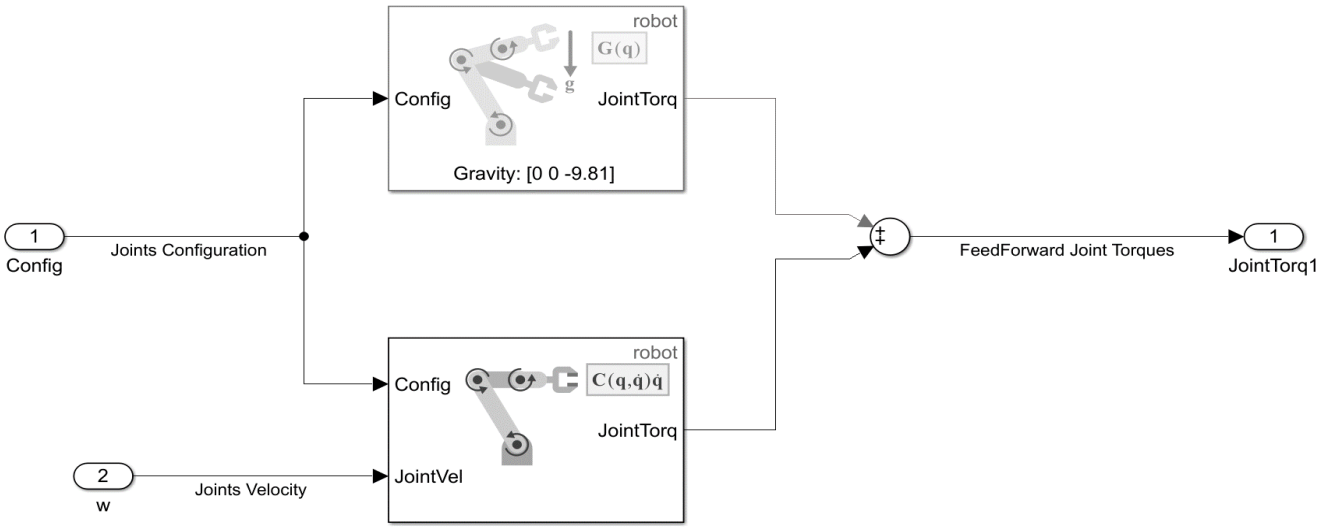


Figure 5.24: Feedforward Block's Diagram

5.2.7. FORWARD KINEMATIC



Figure 5.25: Forward Kinematics Subsystem

In this "Forward Kinematics" subsystem, the position and orientation of the end-effector are calculated. The input of the subsystem is the Joint Configurations, while the output is both the Joint Configurations and the position of the Kuka. The "Get Transform" block is used for the calculation of the transformation matrix for the given configuration, while the "Coordinate Transformation Conversion" block converts this matrix into Euler angles and Translation Vector.

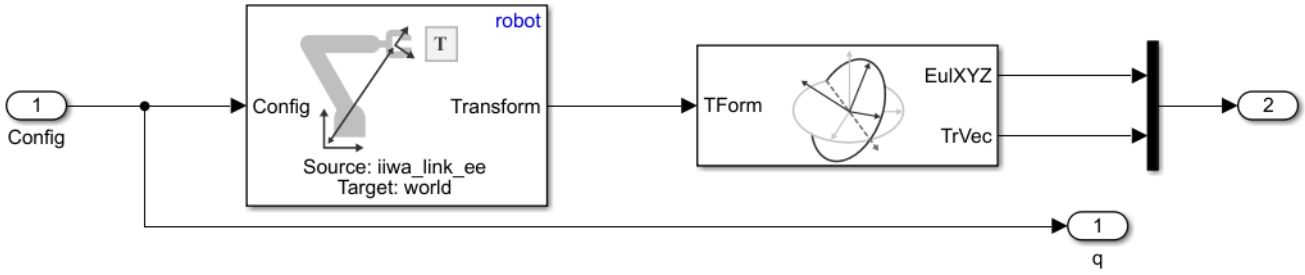


Figure 5.26: Forward Kinematics Diagram

5.2.8. MANIPULATOR FORCES



Figure 5.27: Manipulator's Forces Calculation Block

In this block, forces acting from the manipulator at the end-effector are calculated. The block's inputs are the joint configurations, velocities, accelerations, and joint torques, while the output is a 6x1 vector of the forces and moments. In this block, the Jacobian matrix is initially calculated through the "Get Jacobian" block. The Jacobian of this redundant manipulator is of size 7x6 and so, to be invertible, the first column is removed. In this way, it loses one degree of freedom and specifically of the first joint and the resulting forces are depending on the other six torques. After it is transposed through the "Transpose" block, it is inverted through the line «y=inv(u);» in the "Matlab Function" block.

$$\underline{\boldsymbol{\tau}} = J_{\mathbf{v}}^{T}\underline{\mathbf{F}} \;\rightarrow$$
$$\underline{\mathbf{F}} = (J_{\mathbf{v}}^{T})^{-1}\,\underline{\boldsymbol{\tau}}$$

Joints use some torque to achieve the desired motion and some torque to achieve the desired forces at the environment. If for the calculation of the applied forces the system takes into consideration the joint torques concerning the compensation of the gravity and the acceleration of the end-effector, the resulting values will be incorrect. So, to isolate and remove the joint torques considering the motion of the manipulator, the "Inverse Dynamics" block is used. This block computes the required torques for a given motion given as input. Then, the resulting torques are subtracted from the real joint torques, and the remaining torques are those that create the end-effector's forces. Those are multiplied with the inverse matrix of the transposed Jacobian and the output of the "Product" block are the external forces and the output of the "Manipulator's Forces Calculation" subsystem.

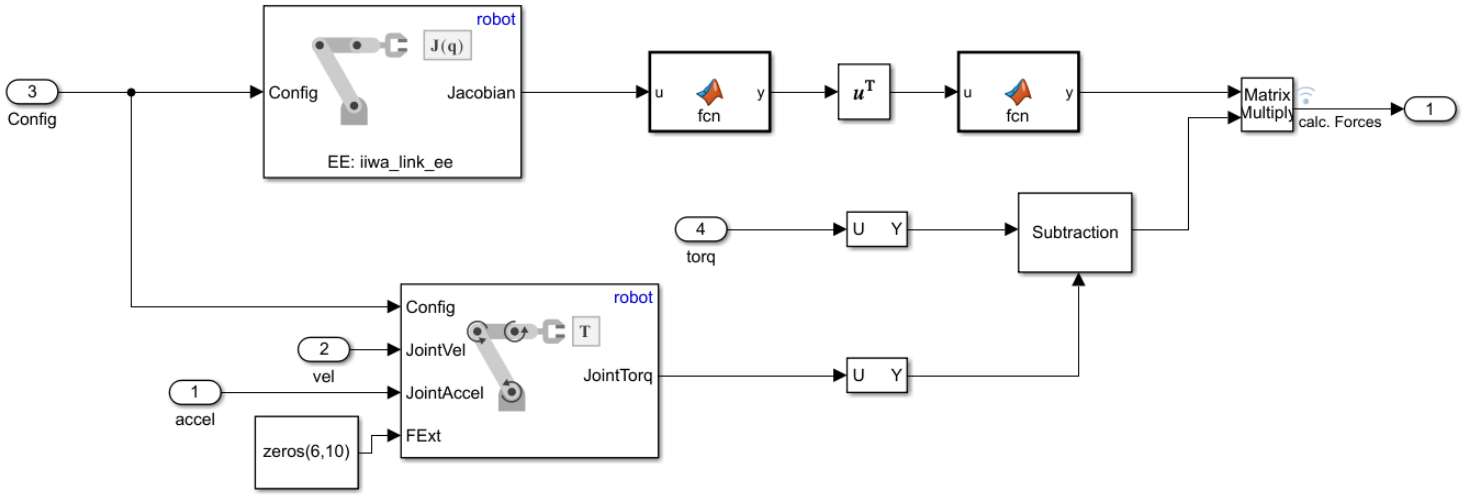


Figure 5.28: Forces Calculator Diagram

5.2.9. DIAGRAMS SUBSYSTEM

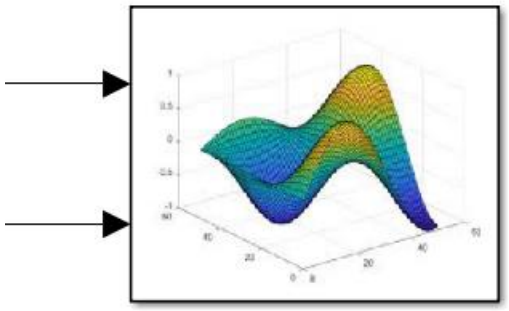


Figure 5.29: Results Subsystem

TRANSPORT VELOCITIES

In this chapter, we introduce the mathematical models that can describe the translational and rotational speed of the end-effector of the robotic arm in relation to the speeds of the joints.

In the relationships that describe the differential changes of the position and the orientation of the end-effector as a function of the differential changes of the positions of the joints, it is usual to use a table called **Jacobian**.

$$\dot{\underline{r}} = J(\underline{\mathbf{q}})\dot{\underline{\mathbf{q}}} \qquad 5.10$$

Where the matrix **J(q)** is not precisely Jacobian, but a different matrix which by convention is called Jacobian or **geometric Jacobian** and is denoted by $\mathbf{J_v(q)}$.

The first three components of this production are the derivatives of the positions, while the other three are the derivatives of Euler angles. The end-effector translational velocities are the direct derivatives of the joints' position, so the ${}^0\underline{v}$ output is the desired one.

$${}^0\underline{V} = \begin{bmatrix} {}^0\underline{v} \\ \hline {}^0\underline{\omega} \end{bmatrix} = {}^0J_v(\underline{q})\dot{\underline{q}} \qquad 5.11$$

In Simulink, this equation comes up with the "Jet Jacobian" block multiplied by the 7x1 joint velocities vector imported in the "Diagram Subsystems" .

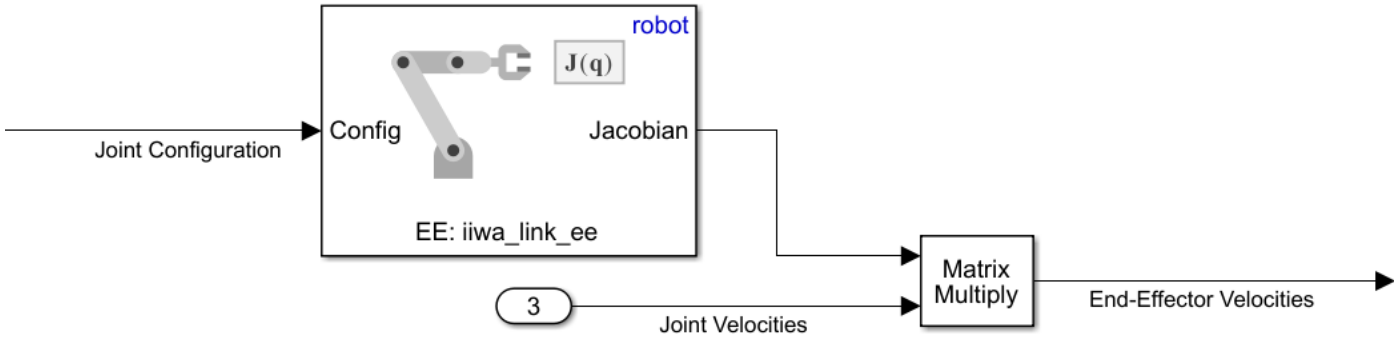


Figure 5.30: End-Effectors Velocities Diagram

ANGULAR VELOCITIES

In contrast, the angular velocity of the end-effector of robotic arms in three-dimensional spaces is not a simple differentiation of Euler angles. A single derivative cannot determine the relationship between the end-effector's angular velocity and the joints' velocity.

$$\underline{\omega} \neq \begin{bmatrix} \dot{r}_4 \\ \dot{r}_5 \\ \dot{r}_6 \end{bmatrix}$$

By algebraic methods and after operations, it has been shown that the rate of change of orientation is related to the derivatives of the Euler angles and is given by the following angular velocity vectors ${}^{U}\omega_{A}$ as follows.

$$\begin{aligned} \omega_x &= c\theta c\psi\dot{\varphi} - s\psi\dot{\theta} \\ \omega_y &= c\theta s\psi\dot{\varphi} - c\psi\dot{\theta} \\ \omega_z &= -s\theta\dot{\varphi} + \dot{\psi} \end{aligned} \qquad 5.12$$

The above equations can be represented in the form of tables as:

$$\underline{\boldsymbol{\omega}} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} c\theta c\psi & -s\psi & 0 \\ c\theta s\psi & c\psi & 0 \\ -s\theta & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \qquad 5.13$$

This equation is represented in Simulink with the diagram shown below. Firstly, the current joint configurations are imported at the "Get Jacobian" block to obtain the homogenous transformation matrix of the manipulator each time. The rotation of the end-effector corresponding to the word coordinate system is given through this matrix. So, it is imported to the "Coordinate Transformation Conversion" block so that the orientation will be exported in the form of Euler's Angles. Here, the φ, θ, and ψ angles represent the rotations about the Z, Y, and X-axis, respectively. Based on the values of these angles each time, the 3x3 matrix and the 3x1 vector of the equation

are formed. The second results from the direct derivative of the vector, while the first is formed through a “Matlab Function”.

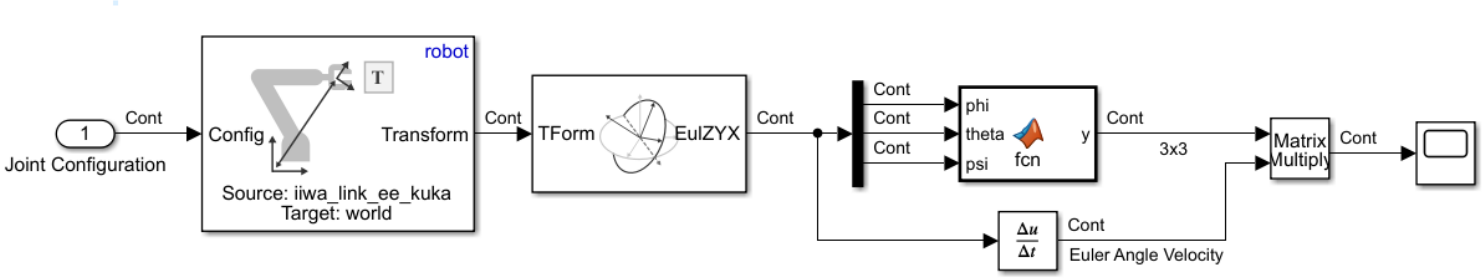


Table 5.2: Rotational Velocity Calculation Diagram

Where the Matlab Function with the φ, θ and ψ inputs creates the 3x3 table shown below:

```
function y = fcn(phi , theta, psi)
u=[  cos(theta)*cos(psi)  , -sin(psi) , 0;...
     cos(theta)*sin(psi)  , cos(psi)  , 0;...
     -sin(theta)          , 0         , 1  ];

y = u;
```

Table 5.3: Rotational Velocity Table

6. RESULTS

First, the applied joint torques of the Kuka are examined so that it will be confirmed that they do not exceed the torque limits of the joint motors. Then the robustness of the controllers will be analyzed based on the position and force error.

6.1. JOINTS TORQUES

First, although the simulation starts at 0 sec, the designed operation takes place after 0.4 sec. This delay has been programmed so that the Kuka is set in a steady state before the operation begins. So, the period from 0 sec, when all the torques are zero, to the time 0.4 sec is not a period of interest.

As shown in the diagrams below, torques actuated by the motors are generally smooth throughout the operation. The only times torques are abruptly changed are at 0.4 and 0.6 sec when the operation pass to another phase. Static frictional forces and the abrupt acceleration can explain that phenomenon. Those abrupt changes of the torques can be seen in the second and fourth motors of the manipulator, as shown in the following figures. The x-axis force applied at the end-effector passes through the z-axis, so it only creates a moment on the y-axis. So, it is reasonable that the abrupt increase of the joint torques occurs at the joints with the rotational joint being on the y-axis, namely the second and fourth joints. Although the sixth joint axis is parallel to the second and fourth, it is almost on the axis of the force, so the moment created is negligible.

Rotational motion concerns the x-axis. Based on the current configuration, the joints that respond at this degree of freedom are mainly the fifth and the seventh. That is why after the time of 0.6 sec, there is a linear increase of the torque at those joints until 1.6 sec when the rotation velocity stabilizes. After that, the torques counterbalance the internal damping of the joint and the external frictional forces due to the screw's rotation.

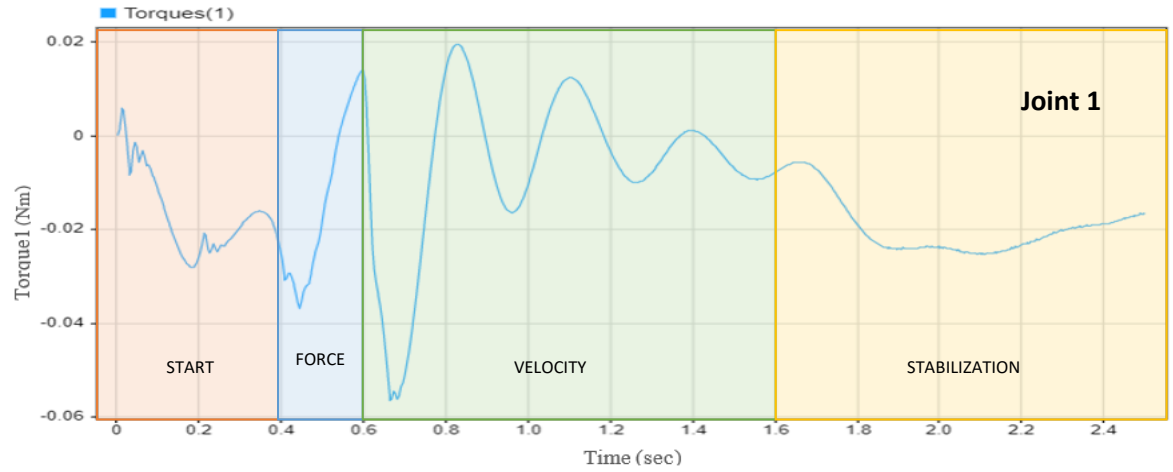


Figure 6.1: Joint1 Torque

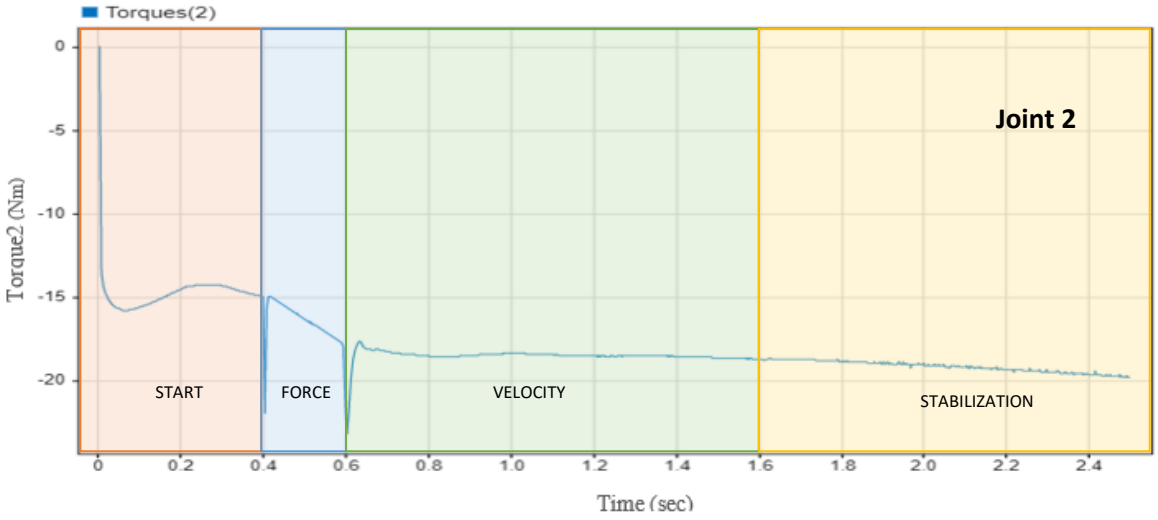


Figure 6.2:Joint2 Torque

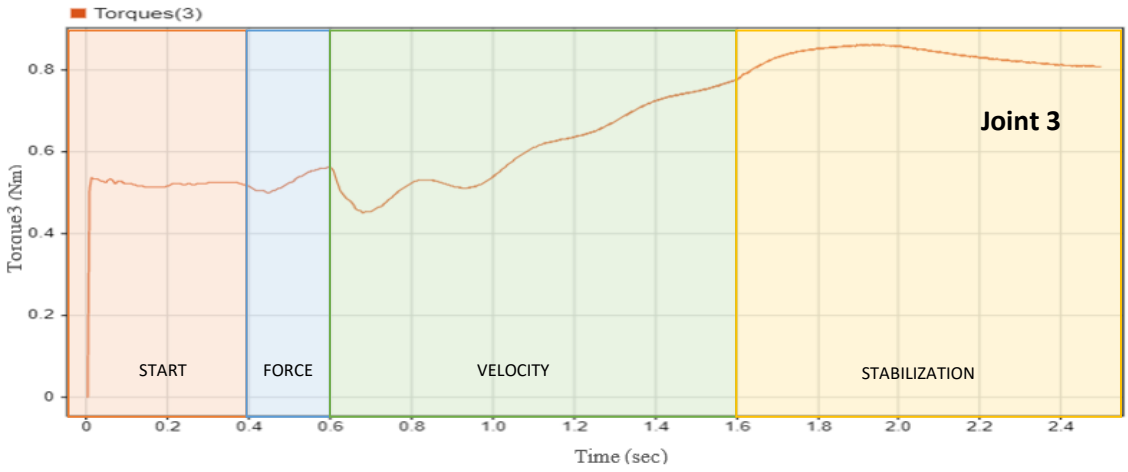


Figure 6.3: Joint3 Torque

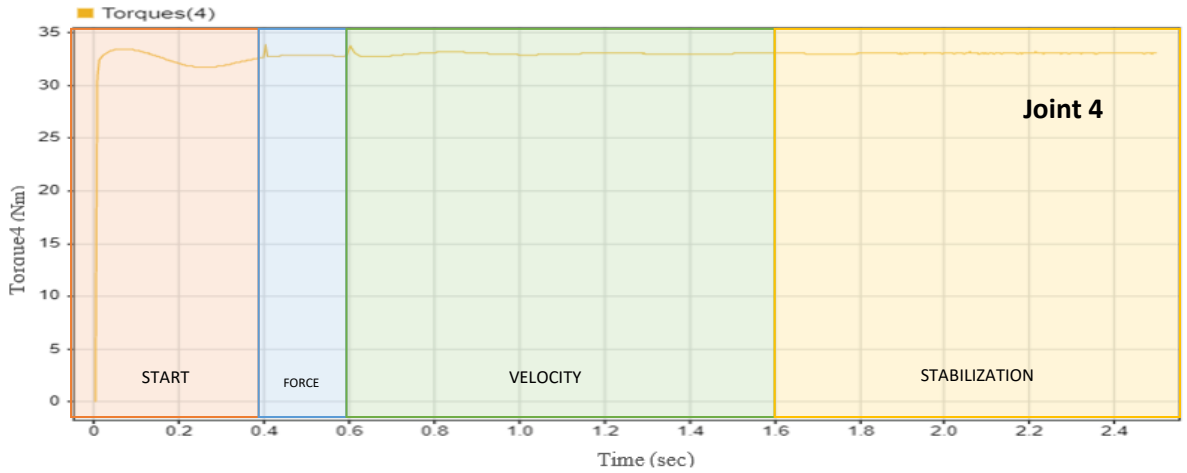


Figure 6.4: Joint4 Torque

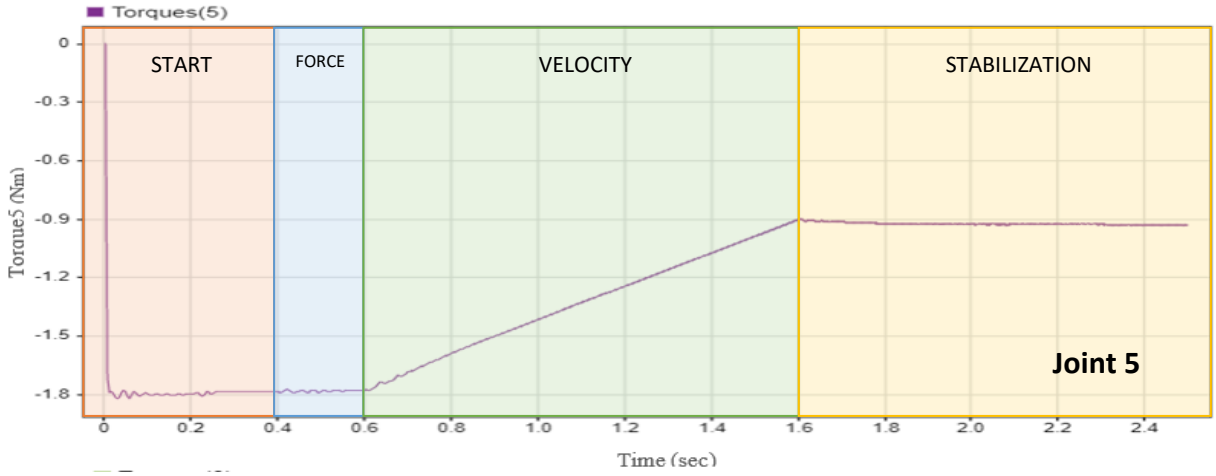


Figure 6.5: Joint5 Torque

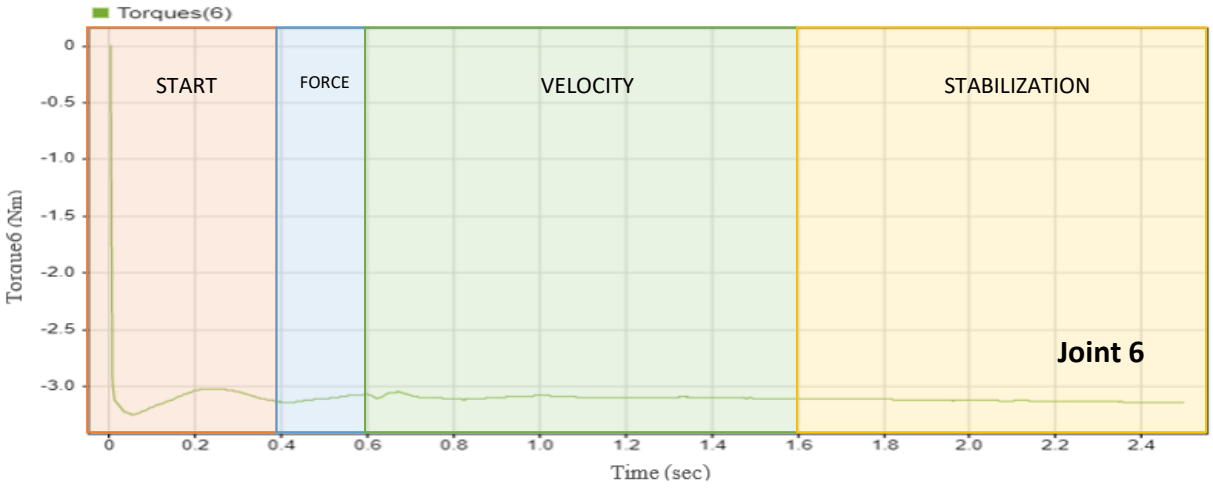


Figure 6.6: Joint6 Torque

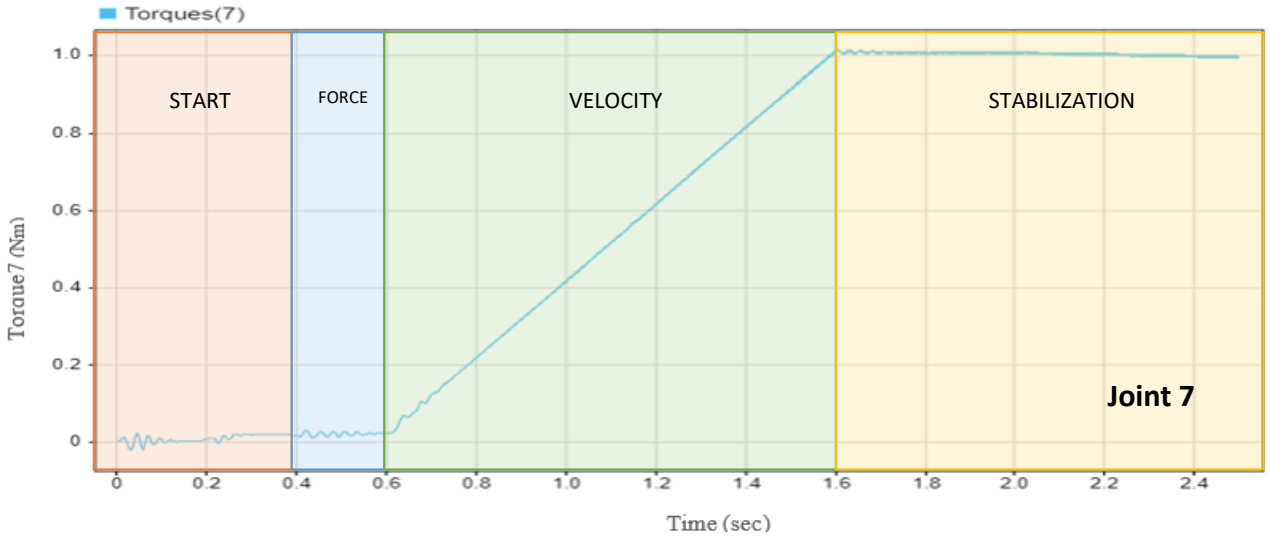


Figure 6.7: Joint7 Torque

The maximum torques at each joint are shown in the following table. As all torques are lower than the limit, the operation is feasible based on the factory properties.

	Joints Bound [Nm]	*Max. Torque [Nm]*
Joint 1	320	0.06
Joint 2	320	19.7
Joint 3	176	0.86
Joint 4	176	33.6
Joint 5	110	1.82
Joint 6	40	3.25
Joint 7	40	1.2

Table 6.1: Joints Bound and Maximum Torques

6.2. END-EFFECTOR POSITION RESPONSE

The position controller is responsible for the end-effector's five out of six degrees of freedom, namely the y and z-axis translational position and the x, y, and z-axis rotational position. Due to the screw's translation and rotation only about the x-axis, the desired y and z position, and the y and z orientation are stable. The rotation about the x-axis accelerates from 0.6 to 1.6 sec till the angle velocity reaches 2 rad/sec, and after that, the velocity stays stable. The desired, as well as the actual positions and rotations, are shown in the plots below.

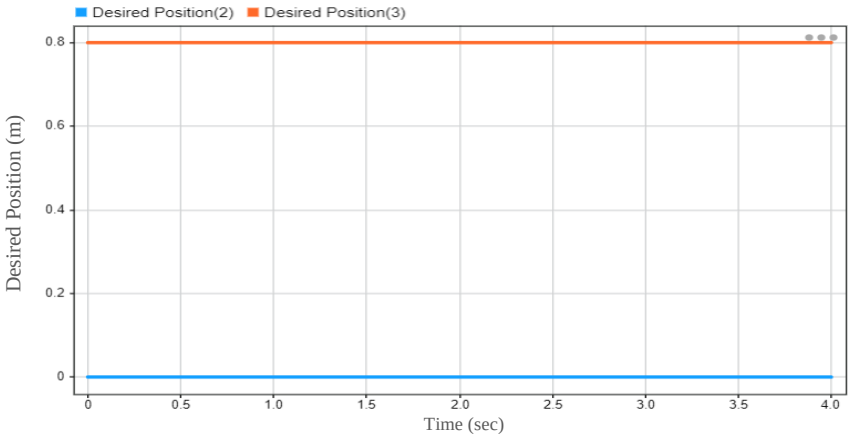


Figure 6.8: Desired Position

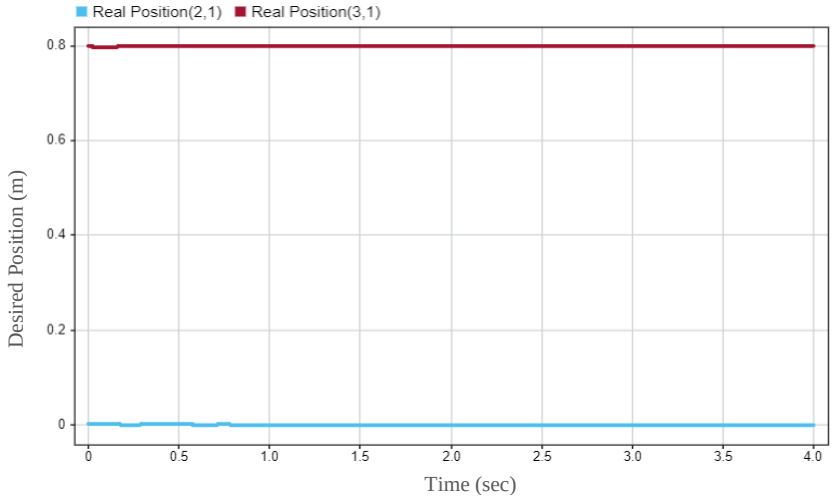


Figure 6.9: Real Position

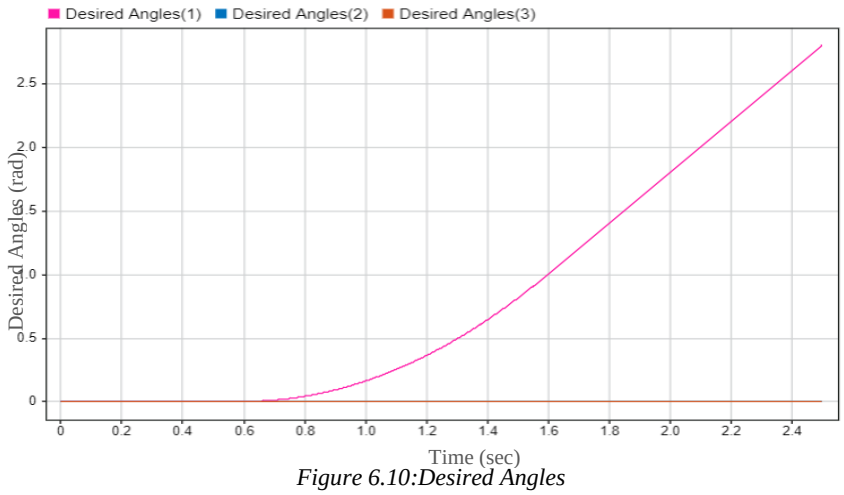


Figure 6.10:Desired Angles

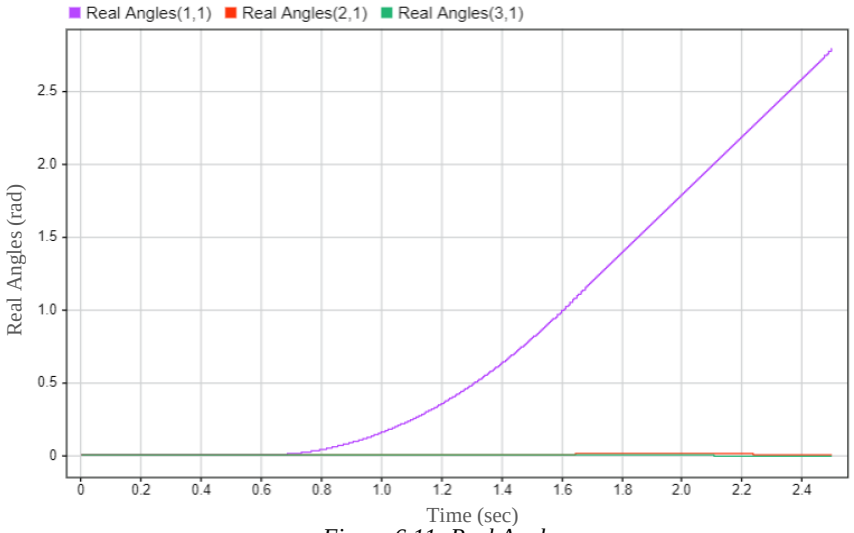


Figure 6.11: Real Angles

As understood from the above figures, the position response of the end-effector is very close to the desired. The plot representing the real angels seems to have a step at the actual angle's value of 3.14, but the boundaries from -π to +π create it.

For the evaluation of the robustnest of the controller, the Position and Orientation Error shown at the next figures is examined.

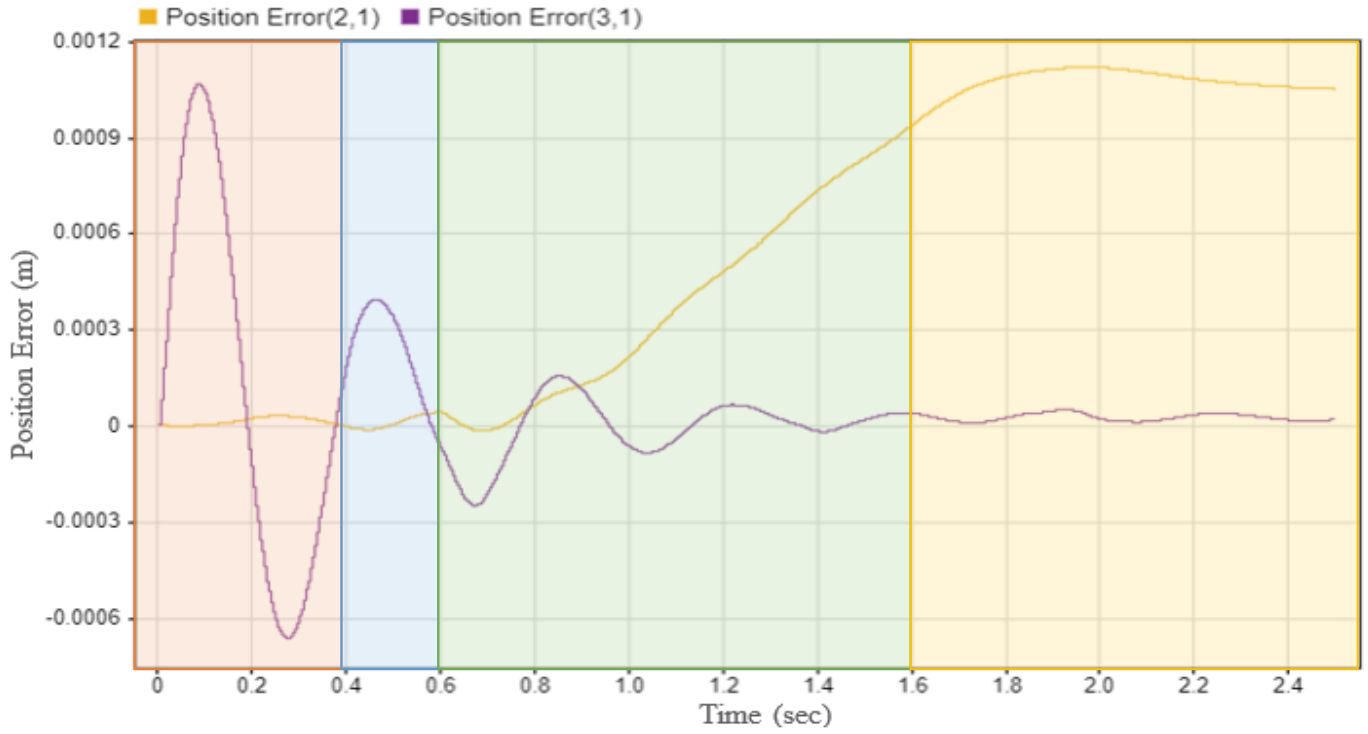


Figure 6.12: Position Error

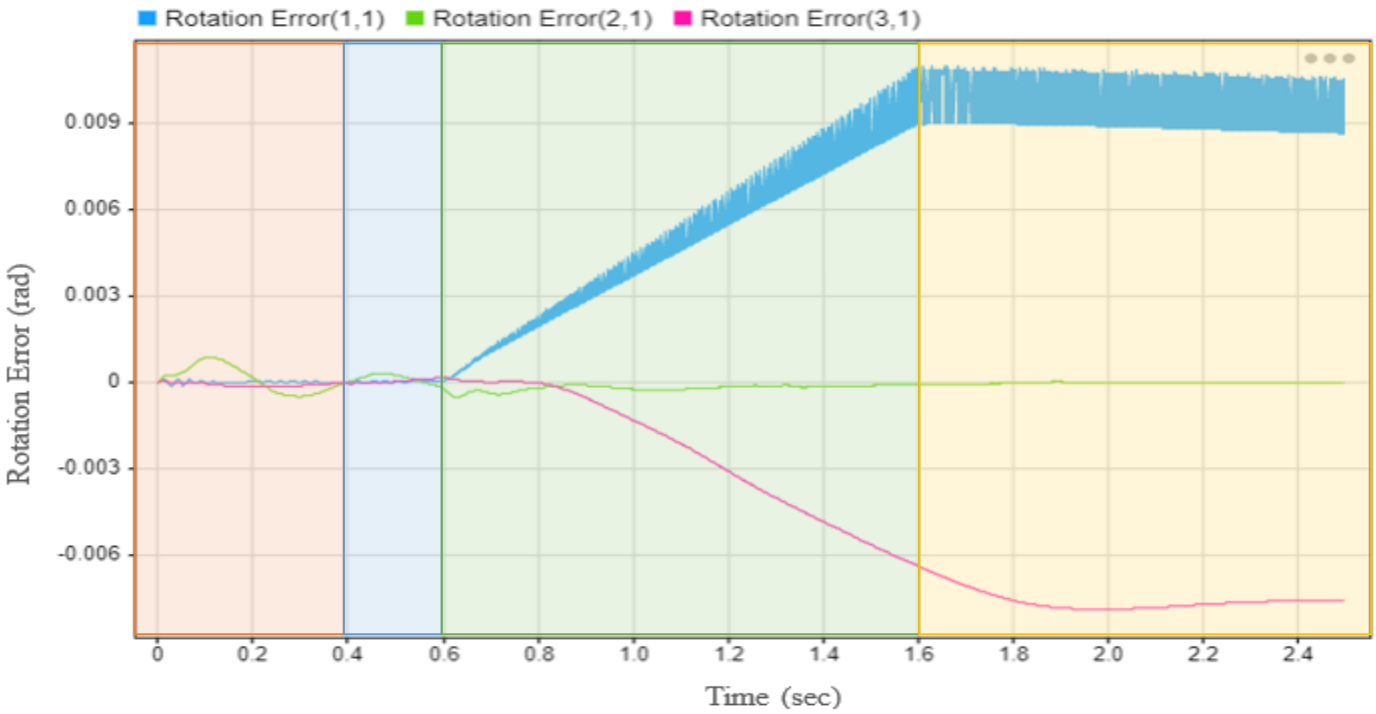


Figure 6.13: Rotation Error

As shown in the error diagrams, the maximum error that existed on the y-axis position is approximately 1.2 mm at the beginning of the simulation at a time of 0.1 sec. Till time 1.5 sec, a damping oscillation is observed, while after that, the error is stabilized under 0.15 mm. The z-axis position on the other size has almost zero error at the beginning, while after 0.8 sec, it is linearly increased. After the peak error of 1.3 mm at 1.8 sec, it starts to fall again, maintaining the error under 1.3 mm. Since the pitch of the screw's thread is about 3 mm, a position error of 1.3 is relatively large.

The maximum error of the orientation, on the other hand, is observed to be 0.01 rad on the x-axis. This axis undergoes the highest orientation alteration compared with the other axes and the highest external moments. The error in the z-axis grows to the value of 0.008 rad, while the error in the y-axis is negligible compared with the other axes' errors.

6.3. END-EFFECTOR FORCE RESPONSE

The force controller section presents the required forces for the designed operation. In the plot diagrams below, the real force is also presented apart from the desired force on the x-axis.

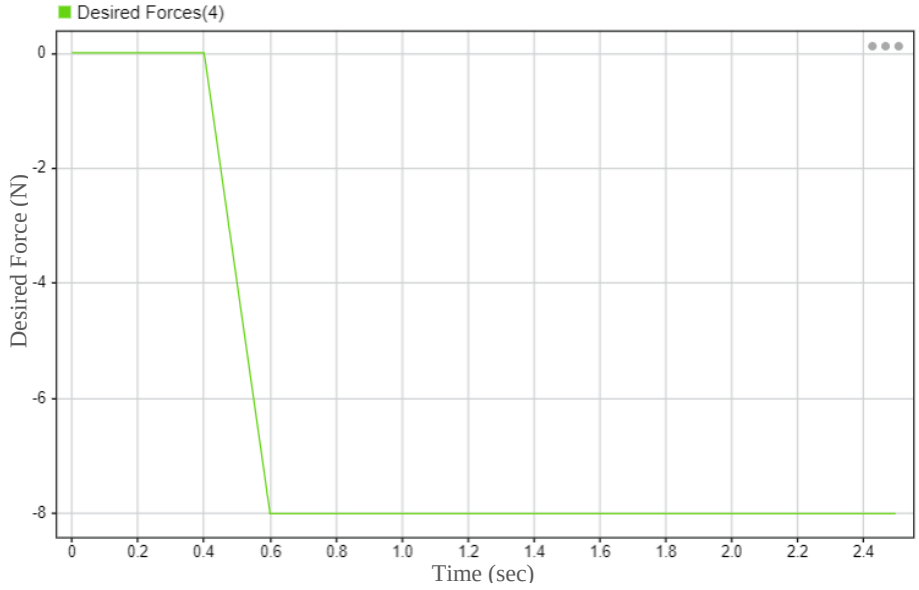


Figure 6.14: Desired Force

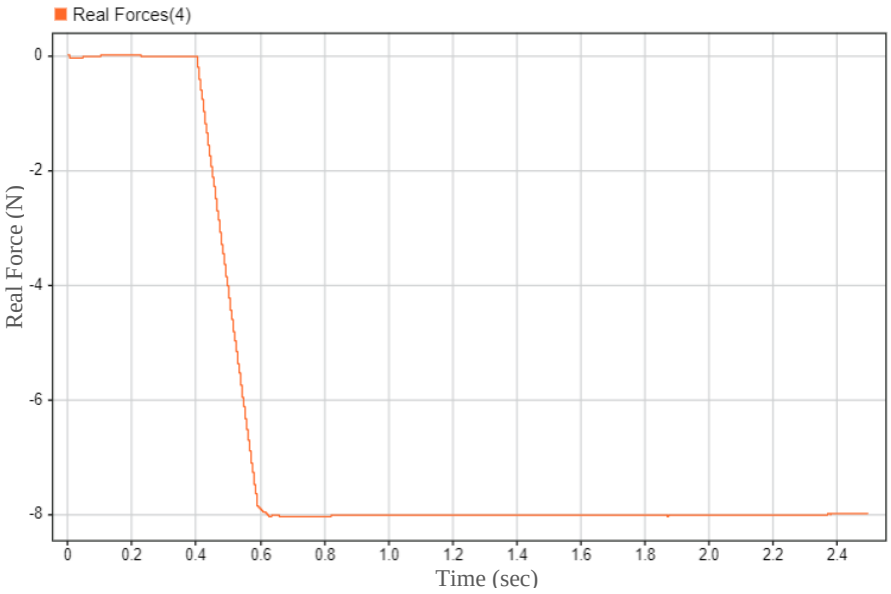


Figure 6.15: Real Force

Apart from the above diagrams, as shown in the error diagram below, the linear increase of the force is achieved satisfactorily. Specifically, the most significant error value exists during the rise of the force, and its value is about 0.25 N. After the stabilization of the desired force at -8 N, the error seems to be less than 0.05 N. At the same time, as time goes on, it continues to decrease.

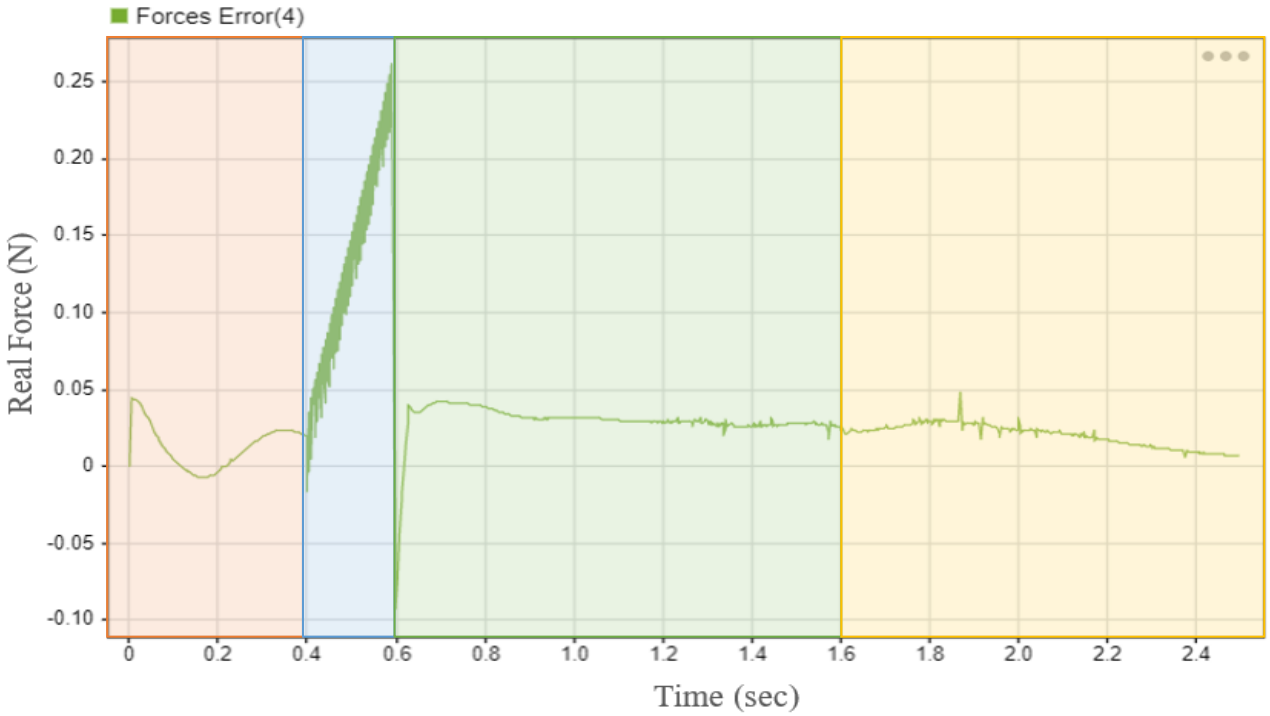


Figure 6.16: Forces Error

7. CONCLUSIONS – DISCUSSION

To complete this research, part of the dynamic hip screw insertion process was simulated through the Ansys finite element analysis program. After the results revealed that the proposed method could be implemented efficiently, the controller of the Kuka IIWA robotic arm for the conduction of the operation was designed.

The method simulated in Ansys aimed to eliminate the rotation of the femoral head during the DHS tightening. Results indicated that the stress developed on the bone was considered acceptable, based on Von-Mise's criteria. At the same time, the sliding distance, and thus the angular displacement, of the femoral head in relation to the femur was almost eliminated. So, the applied method for the insertion of the DHS was validated based on the currently available information. Although, experiments and studies need to be conducted to acquire more precise values of the friction coefficients between bone-bone and bone-screw surfaces.

Moreover, modern methods of obtaining and rendering images on a microscale facilitate the realistic design of the structure of the bone. Thus, at the expense of computational cost, the simulation results can be more reliable, and the reaction forces on the screw would be more precise. This would also facilitate the next simulator, Simulink, to create more accurate results.

During the tightening, no exact values of the forces, torques, and velocities applied to the screw are given. Those values are critical for the effective completion of the operation. However, those values may vary depending on the patient's bone density and the exact spot of screw placement. As presented above, Computer-Assisted Surgery is a growing field of Medicine that personalizes the medical treatments to the patient's personal needs. So, one proposal is to integrate the Finite Element Analysis into the personal preoperative planning of the surgery. Thus, the patient's fracture response can be predicted through the simulator and may be improved through an optimization procedure. This optimization may consider the bones' stress and the femoral head rotation to find the optimal force and angular velocity of the screw for the patient.

Simulation in Explorer Mechanics showed that the end-effector's motion is close to the desired one, which is confirmed by the plotted diagrams in chapter 6. The desired motion of the robotic arm is feasible based on the joint's torque limits. Errors occurred regarding the position and the force of the end-effector, which could be diminished by deepening the PID controllers' design.

The reaction forces were imported to Simulink based on the time of the simulation. In Simulink, there is a time delay in the robot motion due to the PID controllers. So, at a specific time of the Simulink simulation, a frictional force was implemented at the corresponding time of the Ansys simulation. So, this frictional force does not correspond to the manipulator's specific rotational velocity. Maybe, this can be corrected by an iterative method of rerunning the Ansys simulation with the actual forces and velocities resulting from the Simulink. Then the reacting forces of the last FEA will be imported into the Simulink simulation until the error of the actual forces and velocities from the FEA forces and velocities are acceptable.

As mentioned in the first chapter, robotic surgical systems are classified into three categories depending on the surgeon’s interference: autonomous, passive, and semiautonomous. Even though only one part of the process was designed and simulated, the controller of the robotic arm is programmed to accomplish the subprocess of the DHS insertion independently of the surgeon thus it can be categorized as an autonomous system.

Moreover, all the procedures of drilling and threading the bones were considered executed, and the DHS was modeled to be partially already screwed into the femoral head fracture. This thesis can potentially be developed so that an autonomous surgical system will be designed for the entire process of the screw insertion. Maybe, impedance control can be implemented for the initial import of the screw into the opened hole, and supervisory logic be implemented for the termination of the operation when the DHS is fully inserted. For this purpose, kinematic, dynamic, or both data can be used.

BIBLIOGRAPHY

[1] L. J. Melton, “Hip Fractures: A Worldwide Problem Today and Tomorrow,” 1993.

[2] R. H. Taylor and P. Kazanzides, “Medical Robotics and Computer-Integrated Interventional Medicine,” *Advances in Computers*, vol. 73, pp. 219–260, 2008, doi: 10.1016/S0065-2458(08)00405-1.

[3] A. Nasr, A. E. Gaber, and H. A. el Gamal, “Design and Position Control of Arm Manipulator; Experimentally and in MATLAB SimMechanics.” [Online]. Available: www.ijert.org

[4] M. Ben-Ari and F. Mondada, “Elements of Robotics.” ISBN 978-3-319-62532-4 (eBook) https://doi.org/10.1007/978-3-319-62533-1

[5] Π. Εργασία, “Υλοποίηση ρομποτικού βραχίονα με χρήση της πλατφόρμα Arduino.”

[6] J. Rosen et al. (eds.), Surgical Robotics: Systems Applications and Visions, DOI 10.1007/978-1-4419-1126-1_5, # Springer Science+Business Media, LLC 2011

[7] Κούτρα Σοφία, Μπαλάφα Ιωάννα “Εφαρμογές ρομποτικής χειρουργικής και ο ρόλος εξειδικευμένου νοσηλευτή”, Πτυχιακή Εργασία, Νοσηλευτική, Τεχνολογικό Εκπαιδευτικό Ίδρυμα Δυτικής Ελλάδας, Σχολή Επαγγελμάτων υγείας και Πρόνοιας, Τμήμα Νοσηλευτικής.

[8] J. H. Lonner and H. Arthroplasty, “Robotics in Knee and Hip Arthroplasty: Current Concepts, Techniques and Emerging”, Uses 1st ed. 2019 Edition, Kindle Edition

[9] D. W. Skarecky, “Robotic-Assisted Radical Prostatectomy after the First Decade: Surgical Evolution or New Paradigm,” *ISRN Urology*, vol. 2013, pp. 1–22, Apr. 2013, doi: 10.1155/2013/157379.

[10] Q. D. Trinh *et al.*, “Perioperative outcomes of robot-assisted radical prostatectomy compared with open radical prostatectomy: Results from the nationwide inpatient sample,” *European Urology*, vol. 61, no. 4, pp. 679–685, Apr. 2012, doi: 10.1016/j.eururo.2011.12.027.

[11] Loulmet D, Carpentier A, d'Attellis N, Berrebi A, Cardon C, Ponzio O, Aupècle B, Relland JY. Endoscopic coronary artery bypass grafting with the aid of robotic assisted instruments.

J Thorac Cardiovasc Surg. 1999 Jul;118(1):4-10. doi: 10.1016/S0022-5223(99)70133-9. PMID: 10384177.

[12] A. Ghasem, A. Sharma, D. N. Greif, M. Alam, and M. al Maaieh, “The arrival of robotics in spine surgery,” *Spine*, vol. 43, no. 23. Lippincott Williams and Wilkins, pp. 1670–1677, 2018. doi: 10.1097/BRS.0000000000002695.

[13] G. Molliqaj *et al.*, “Accuracy of robot-guided versus freehand fluoroscopy-assisted pedicle screw insertion in thoracolumbar spinal surgery,” *Neurosurgical Focus*, vol. 42, no. 5, 2017, doi: 10.3171/2017.3.FOCUS179.

[14] Guoyan Zheng, Wei Tian, Xiahai Zhuang, “Intelligent Orthopaedics: Artificial Intelligence and Smart Image-guided Technology for Orthopaedics”, 2018, ISBN 978-981-13-1396-7 (eBook), https://doi.org/10.1007/978-981-13-1396-7

[15] N. Sugano, “Computer-assisted orthopaedic surgery and robotic surgery in total hip arthroplasty,” *Clinics in Orthopedic Surgery*, vol. 5, no. 1. pp. 1–9, Mar. 2013. doi: 10.4055/cios.2013.5.1.1.

[16] Pransky, J. (1997), "ROBODOC - surgical robot success story", Industrial Robot, Vol. 24 No. 3, pp. 231-233. https://doi.org/10.1108/01439919710167444

[17] J. E. Lang *et al.*, “Robotic systems in orthopaedic surgery,” *THE JOURNAL OF BONE AND JOINT SURGERY ν SPECIALTY UPDATE: GENERAL ORTHOPAEDICS J Bone Joint Surg Br*, pp. 93–1296, 2011, doi: 10.1302/0301-620X.93B10.

[18] Sugano N. Computer-assisted orthopedic surgery. J Orthop Sci. 2003;8(3):442-8. doi: 10.1007/s10776-002-0623-6. PMID: 12768493

[19] R. A. States and E. Pappas, “Precision and repeatability of the Optotrak 3020 motion measurement system,” *Journal of Medical Engineering and Technology*, vol. 30, no. 1, pp. 11–16, Feb. 2006, doi: 10.1080/03091900512331304556.

[20] J. P. van der List, H. Chawla, and A. D. Pearle, “Robotic-Assisted Knee Arthroplasty: An Overview.” [Online]. Available: www.amjorthopedics.com

[21] J. H. Lonner, “Robotically Assisted Unicompartmental Knee Arthroplasty with a Handheld Image-Free Sculpting Tool,” *Operative Techniques in Orthopaedics*, vol. 25, no. 2, pp. 104–113, Jun. 2015, doi: 10.1053/j.oto.2015.03.001.

[22] C. Batailler, D. Hannouche, F. Benazzo, and S. Parratte, “Concepts and techniques of a new robotically assisted technique for total knee arthroplasty: the ROSA knee system,” *Archives of Orthopaedic and Trauma Surgery*, vol. 141, no. 12. Springer Science and Business Media Deutschland GmbH, pp. 2049–2058, Dec. 01, 2021. doi: 10.1007/s00402-021-04048-y.

[23] “Omega3 System. Compression Hip Screw. Operative Technique”, Stryker 1 Bohnackerweg Selzach 2545-CH Switzerland, 11-2015

[24] B. R. MICHAEL BAIJMGAERTNER MDt STEPHEN L CURTIN MDt DIETER M LINDSKOG BAt, “The Value of the Tip-Apex Distance in Predicting Failure of Fixation of Peritrochanteric Fractures of the Hip*.”

[25] A. Mulay, *Clinical Case Study: Spine Modeling for Minimum Invasive Spine Surgeries (MISS) using Rapid Prototyping*. 2018. [Online]. Available: https://www.researchgate.net/publication/323377607

[26] J. S. Shockey, J. A. von Fraunhofer, and D. Seligson, “A MEASUREMENT OF THE COEFFICIENT OF STATIC FRICTION OF HUMAN LONG BONES,” 1985.

[27] X. Gao, M. Fraulob, and G. Haïat, “Biomechanical behaviours of the bone-implant interface: A review,” *Journal of the Royal Society Interface*, vol. 16, no. 156. Royal Society Publishing, 2019. doi: 10.1098/rsif.2019.0259.

[28] J. Li *et al.*, “Finite element analysis of different configurations of fully threaded cannulated screw in the treatment of unstable femoral neck fractures,” *Journal of Orthopaedic Surgery and Research*, vol. 13, no. 1, Oct. 2018, doi: 10.1186/s13018-018-0970-3.

[29] J. Li, P. Yin, L. Zhang, H. Chen, and P. Tang, “Medial anatomical buttress plate in treating displaced femoral neck fracture a finite element analysis,” *Injury*, vol. 50, no. 11, pp. 1895–1900, Nov. 2019, doi: 10.1016/j.injury.2019.08.024.

[30] J. M. Goffin, P. Pankaj, and A. H. Simpson, “A computational study on the effect of fracture intrusion distance in three- and four-part trochanteric fractures treated with Gamma nail and sliding hip screw,” *Journal of Orthopaedic Research*, vol. 32, no. 1, pp. 39–45, Jan. 2014, doi: 10.1002/jor.22469.

[31] T. M. Keaveny, E. F. Morgan, G. L. Niebur, and O. C. Yeh, “BIOMECHANICS OF TRABECULAR BONE,” 2001. [Online]. Available: www.annualreviews.org

[32] Denavit J. and R. S. Hartenberg, “A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices.,” *ASME. J. Appl. Mech.*, pp. 215–221, Jun. 1955.

[33] I. S. M. Khalil, A. Alfar, A. F. Tabak, A. Klingner, S. Stramigioli, and M. Sitti, “Positioning of drug carriers using permanent magnet-based robotic system in three-dimensional space,” in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, Aug. 2017, pp. 1117–1122. doi: 10.1109/AIM.2017.8014168.

[34] S. Fang, X. Ma, Y. Zhao, Q. Zhang, and Y. Li, “Trajectory Planning for Seven-DOF Robotic Arm Based on Quintic Polynormial,” in *Proceedings - 2019 11th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2019*, Aug. 2019, vol. 2, pp. 198–201. doi: 10.1109/IHMSC.2019.10142.

[35] Luh J., Walker M., and Paul R., “On-line Computational Scheme for Mechanical Manipulators,” *J. Dynamic Systems, Measurement*, pp. 69–76, Jun. 1980.

[36] ΑΣΠΡΑΓΚΑΘΟΣ ΝΙΚΟΣ, *ΡΟΜΠΟΤΙΚΗ Μηχανική, Έλεγχος και Σχεδιασμός Κίνησης*. ΠΑΤΡΑ, 2018.