UNIVERSITÀ POLITECNICA DELLE MARCHE
FACOLTÀ DI INGEGNERIA

Corso di Laurea in Biomedical Engineering

Definition of an upper limb rehabilitation protocol using a collaborative robot

Advisor:
Prof. Ing. Giacomo Palmieri

Student:
Maria Letizia Dini

A.A. 2020/2021

Acknowledgments

A sincere thanks to my supervisor Mr. Palmieri for his invaluable advices and for the great opportunity he gave me to carry out the experimental part of this project. Thanks also to Giorgia, who guided me along this path.

Thanks to my family who have always supported me.

Thanks to Andriy for encoraging me to never give up. You are my dearest example.

Thanks to all my friend, to those closest to me who have never abandoned me and to those furthest away, who manage to make me feel their love from miles away.

Abstract

The thesis project represents a preliminary study on data from subjects under treatment at the neuro-motor rehabilitation department of the Torrette Hospital. The investigation constitutes an experimental study on one of the robots available within the Politecnica delle Marche University, namely Universal Robot (UR5e). The aim of this study is to evaluate whether this type of cobot is suitable for rehabilitation therapy of upper limbs disease. The thesis is structured in 5 main chapters: in the first and in the second, background and clinical context are presented, and the rationale and purpose of the thesis are explained; the third chapter presents the materials and methods used, the fourth chapter presents the results obtained during the study. Finally, the last chapter contains the conclusions emerging from the work developed, and the hints for a possible subsequent deepening of the investigation.

Contents

CHAPTER 1

INTRODUCTION

The work presented in this thesis is the first part of a research whose final aim is to identify the key mechanisms for determining patient involvement during post-trauma assisted robotic therapy, to optimize the design of robotic devices. The main hypothesis behind the research is that the patient's involvement and effort are related to the sensory information provided by the robotic system, and the more the patient will be involved, the more there will be increases in the benefits of assisted robotic therapy. To achieve these primary results, a latest generation robot from Universal Robots, a leading company in the world market, has been programmed: e-series UR5. During this project, a type of rehabilitation exercise was programmed in the language of the robot, in which the patient, holding on to the ergonomic handle implemented in the robotic system, must drive until a target is reached, displayed in the work area. The final aim of this project will be to implement and modify this type of therapeutic exercise, to finally model a general method in which the patient interacts with the robot. In this way it will be possible to understand the patient's response to a level sufficient to dictate guidelines in the design of future robotic devices. A key point will be to define the variables used to quantify patient participation and sensory inputs in the computational model, and their measurement method. To extrapolate this type of information, a cross-questionnaire was defined for patient satisfaction and UR Log Viewer, a software provided by the company itself, and a study on the forces developed by the robot and the patient were implemented. Clinical trials were conducted with 19 subjects, including healthy subjects (as a control group) and post-trauma patients. The test results confirmed the initial hypotheses: large differences between the traumatized and non-traumatized sides in patients and differences between the control group and post-trauma patients. A subsequent step of the thesis work will concern the improvement of the robot-patient interface and the computational motor control model for post-trauma patients.

CHAPTER 2

STATE OF ART

2.1 Mechanism of the upper limb

The joint mechanics of the shoulder joint is very complex, because being made up of several joints, it has a great variety of movements. In fact, this joint is the most mobile in the human body [1]. It has three degrees of movement that allow the orientation of the upper limb in relation to the three planes of space thanks to its three main axes:

- Transverse axis, contained in the frontal plane allows flexion-extension movements performed in a sagittal plane (180°-50°);

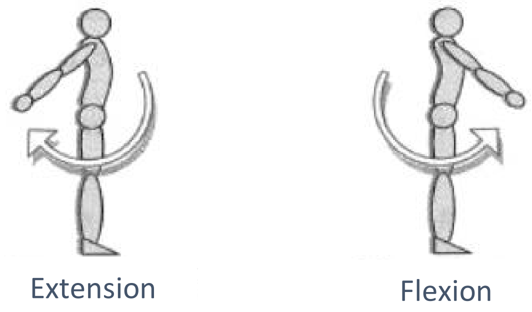


Figure 1 – Flexion and extension movements of the shoulder.

- Antero-posterior axis, contained in the sagittal plane allows abduction and adduction movements carried out in a frontal plane (180°-30°);

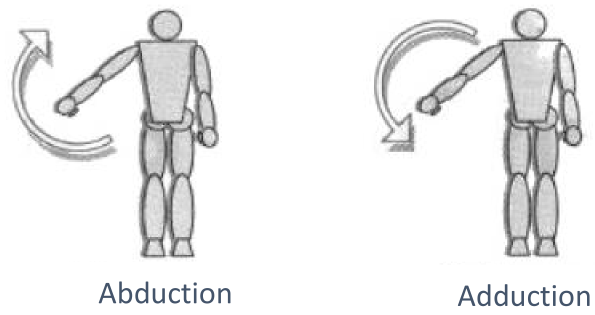


Figure 2 - Abduction and adduction movements of the shoulder.

o Vertical axis, determined by the intersection of the two previous planes: it allows flexion and extension movements performed in a horizontal plane, keeping the arm abducted at 90° (140°-30°).

Figure 3 – Internal and external rotations

The point where the three median planes of the body intersect corresponds to the centre of gravity.

By combining the elementary movements around the three axes, then, in succession, flexion, abduction, extension and adduction, the circling movement is described. During the movement, the arm describes in space an irregular cone called: cone of circumvention. The latter defines, in a sphere centred on the shoulder and a radius equal to the length of the upper limb, a spherical sector of accessibility, within which the hand can reach objects without moving the trunk.

The observation of movement and the time when it deviates from the normal canons can be a fundamental clinical element for the diagnosis: understanding which muscle or district to attribute the functional deficit indicates the way for a targeted and effective rehabilitation treatment [2] [3].

2.1 Rehabilitation Upper Limbs

It is estimated that about 2.4 billion people worldwide are currently living with a health condition that benefits from rehabilitation. With modifications taking place in the health and characteristics of the global population, this estimated need for rehabilitation is only going to rise in the coming years. The reason is that people are living longer, with the amount of people over 60 years of age predicted to double by 2050 and more of them will live with chronic diseases such as diabetes, stroke, and cancer. Also present a high rate of incidence of injury and cerebral disfunctions. For these conditions rehabilitation can be beneficial. In many parts of the world, this increasing need for rehabilitation is going largely unmet. More than half of people living in some low- and middle-income countries who require rehabilitation services do not receive them.

Rehabilitation following most neurological trauma largely consists in carrying out activities related to the actions of daily life or aimed at achieving a goal (goal-directed training), in which the patient is required to perform repeated movements to produce functional motor patterns. The goal of the rehabilitation intervention is to achieve the best level of patient autonomy in the acute and sub-acute phases (up to about 1 year after the event), and to maintain and consolidate the results achieved by physiotherapy in the subsequent period. Following a lesion of the central nervous system, part of the autonomy and the ability to carry out the activities of daily life are lost; the neurological damage, usually consequent to ischemic (95%) or haemorrhagic events, determines a cellular necrosis which is precisely the cause of the neuro-motor deficit. In the clinical setting, the phenomenon is known as neuroplasticity or brain plasticity (which is the basis of the consequent motor re-learning). This behavior can be stimulated, strengthened, and guided through a rehabilitation process, with the aim of decreasing the effects of the brain injury, therefore the degree of disability, and strengthening residual functions in the best possible way. The most common consequence of stroke, which affects about 80% of surviving patients, is motor disability [4]: in most cases it results in total or partial incapacity to control muscles and movements of the face and limbs. The function of the upper limb is impaired in the acute phase in 85% of patients. In the following 3-6 months a variable percentage between 55 and 75% of cases still show this interest. Intensive treatment can improve function in subjects of medium severity at 6 months, while there is no effect in severe subjects [5]. Given the importance of the problem and given the fact that the significant progress made in recent times in the field of motor rehabilitation have greatly reduced the number of outcomes of neurolysins considered irreversible until recently, a great deal of attention has been paid to the rehabilitation process.

During these decays, the field of new technologies dedicated to upper-limb rehabilitation is exploding. This field has been developing rapidly over the years with an increasing number of publications regarding robots, virtual reality and telerehabilitation for neurorehabilitation of the upper limb. The existing models of neurological rehabilitation approaches to promote motor recovery is dedicated on repetitive, high-intensity and task-specific practice. Technological tools are well suited to fulfil these rehabilitation principles, as they can be controlled by users, repeatable, intensive, interactive, and motivating rehabilitation with feedback. The amount and variety of robots being created by teams all over the world is increasing continuously. It has been widely demonstrated that the use of technological devices for training could be a useful adjunct to therapy, since robot can provide repetitions and the same intensity, without over-burdening therapists. Furthermore, the assistance provided to the patient by the instrument "releases" the therapist in some way, allowing him to concentrate on other, more critical, aspects of rehabilitation or even to administer the treatment on more than one individual simultaneously. The combination of a rehabilitation robot with exercises carried out in a virtual environment also appears potentially beneficial in terms of involvement and participation on the part of the patient which, as has been said, are fundamental factors for the recovery of the motor and functional skills of the limb [6]. From the literature, there are many examples of robotic upper limb therapy devices include the MIT-Manus [7], l' ARM Guide [8], MIME [9] and others.

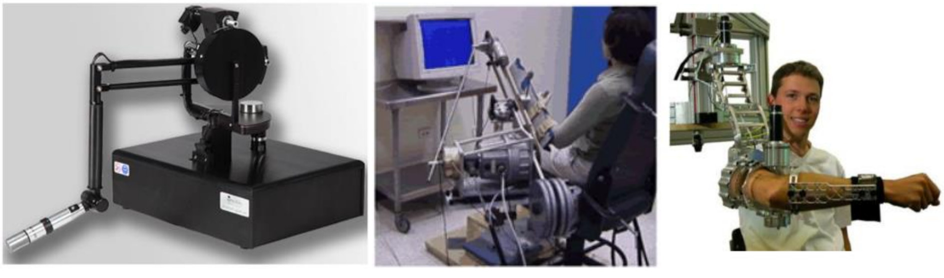


Figure 4 – Examples of robotic upper limb rehabilitation devices: MIT-Manus, ARM Guide, MIME.

The aim of this study is to introduce a new device, the Cobot UR5e, of ©Universal Robots, as a tool to realize upper limb rehabilitation. The innovation is that is the first in this field, since UR bots are industrial-type and are intended for the use of final tools / actuators and equipment or for the processing or transfer of components or products.

2.2 Universal Robots

The Universal Robot (UR) is a company that was founded in 2005 and has developed three models of machines: UR3, UR5 and UR10. Unlike traditional robots, which are big and expensive, this new generation of cobots is intended to be easy to program and safe for every cooperation activity. These cobots can be used in many industrial employments and the choice of the best matched UR model depends on the type of work and on the possible load to be moved. Recently, the UR company has created a new series of robots by improving the earlier one: e-series cobots (eUR3, eUR5, eUR10). In Table 1 are reported all the main characteristics of these three cobots.These new cobots are equipped with more intuitive programming and additional components to be employed in multiple industry production. They are more versatile with an increased number of safety functions. Each robot of UR is named after their payload capacity (kilograms).

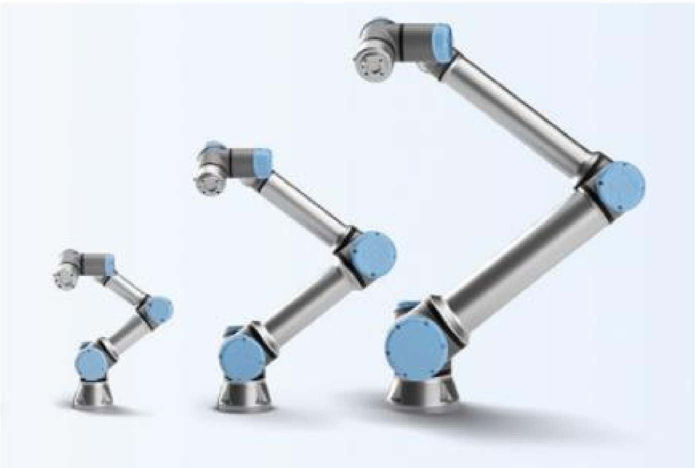


Figure 5 - UR3e, UR5e and UR10e Cobots.

		UR3	UR5	UR10
Weight [Kg]		11	18.4	28.9
Payload [Kg]		3	5	10
Reach [mm]		500	850	1300
Joint ranges [°]		±360	±360	±360
Speed	Base joint	±180°/s	±180°/s	±120°/s
	Shoulder joint	±180°/s	±180°/s	±120°/s
	Elbow joint	±180°/s	±180°/s	±180°/s
	1st wrist joint	±180°/s	±180°/s	±180°/s
	2nd wrist joint	±180°/s	±180°/s	±180°/s
	3rd wrist joint	±180°/s	±180°/s	±180°/s
	Tool	1m/s	1m/s	1m/s

Table 1 - Main characteristics of the UR cobots

The most used in rehabilitation process is the UR5, that is marginally bigger than UR3 and can reach a perfect relationship between size and power. It has an internal controller safety mechanism that assures a corrected force control approach. This mechanism stops all movements of the manipulator within 500 ms if the force action on the tool-centre-point (TCP), the centre point of the output flange, exceeds 25 Kem/s. UR5 is ideal for automating low-weight processing tasks (up to 5 Kg) like picking, placing, and testing. It is capable to reach radius up to 850 mm from the base joint (robot workspace) and combine compact measures (about 20 Kg with a footprint of 149 mm).

As shown in Figure 6, the robot is composed by an arm with 6 degree of freedom and six joints:

- base,
- shoulder,
- elbow,
- wrist 1, wrist 2 and wrist 3.

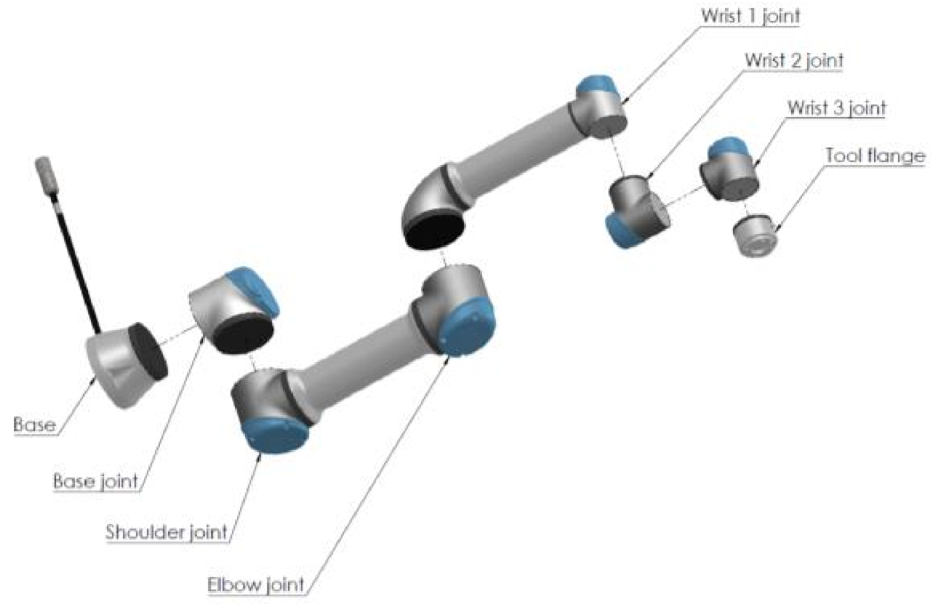


Figure 6 - Main components of UR5e

The electrical equipment of UR5e is composed by:

- a controller, for digital and analogue control signals;
- a connector at the tool end that delivers power and control signals for gripper and sensors;
- an Ethernet connection, provided at the bottom of the control box.

The Universal Robot can be controlled at two levels:

- The PolyScope or the Graphical User Interface Level
- Script Level

At the Script Level, the URScript is the programming language that controls the robot. The URScript includes variables, types, and the flow control statements. There are also built-in variables and functions that monitor and control I/O and robot movements. Thanks to touchscreen, called teach pendant, programming is quick and easy. The robot can be programmed with UR script language or directly through the interface using the pad .

So, the choice of using UR5e to develop a rehabilitation exercise is given by all these characteristics just mentioned above.

CHAPTER 3

PROGRAMMING UR5e
FOR REHABILITATION TASK

In this chapter, it will be present how was built the workbench and how it is composed. Moreover, the rehabilitation script will be explained and finally results will be showed.

3.1 Workbench

In the Figure 11 it is showed the main components of the workbench:

- Bench, where camera system and cobot UR5 lays. The bench measured 90cm x 80cm, and 80cm in height. The bench was covered by black pannel, to avoid any light interferences for the camera. It was also defined an area, (the white one) of 30cm x 25cm, that defines the working range of the camera. In this region, target can be founded.

- Smart Camera System (Cognex In-Sight 7600). The camera was integrated as The camera recognizes the target and updates the coordinates in the script every time the target changes position.

- Target. It is strictly necessary if the script wants to run. The position of the target is updated and every time the program can calculate the trajectory from the TCP to the target centre. However, it is possible to implement different target just after being recognized from the camera. The target chosen in the one represented in Figure 7.

- Chair, where subjects can stay and perform the exercise.

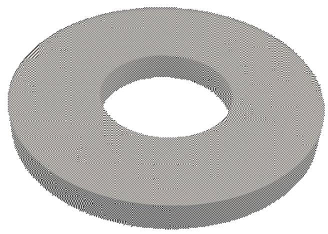


Figure 7 - Target.

- Ergonomic handle: an ergonomic handle was designed in ©Catia, the leading 3D CAD design software developed by the French Dassault Systems, with an extremely usable interface for any type of user. This ergonomic handle measures 20cm and it is showed in Figure 8. This handle help patient to facilitate the grip during the exercise. Moreover, in more serious cases, it was implemented a fixing glove, suitable for those patients who do not have the strength to grasp.

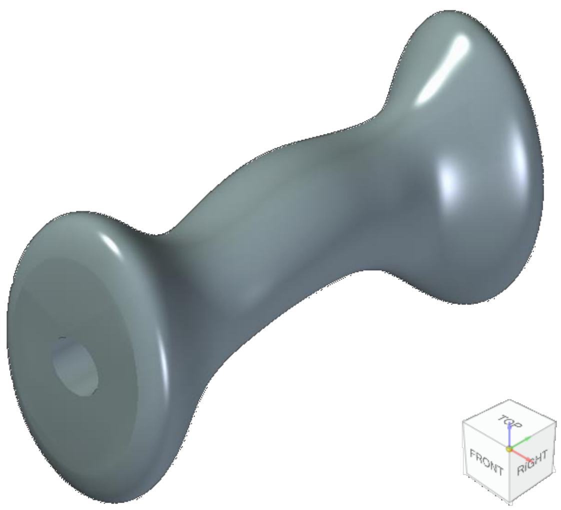


Figure 8 - 3D model of the ergonomic handle.

The ergonomic handle was implemented with a tip (Figure 9), positioned on the bottom side of the handle. This tip measures 3cm in diameter and 3cm in height and was designed on Catia 3D CAD. The need of this components is to help patients to arrive at the specific target.

Figure 9 - Model Tip

- Mallory Son alert PF-35N29WQ: every time one task is completed a sound is emitted.
  - Single 'bip' indicates the different measurement steps
  - Double 'bip' indicates if the target is reached
  - Triple 'bip' indicates that the one modality exercise is over

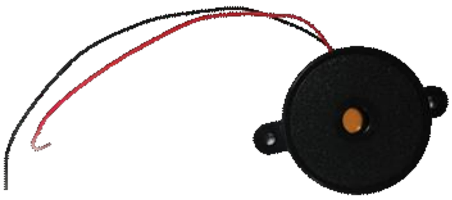


Figure 10 - Mallory Son alert PF-35N29WQ

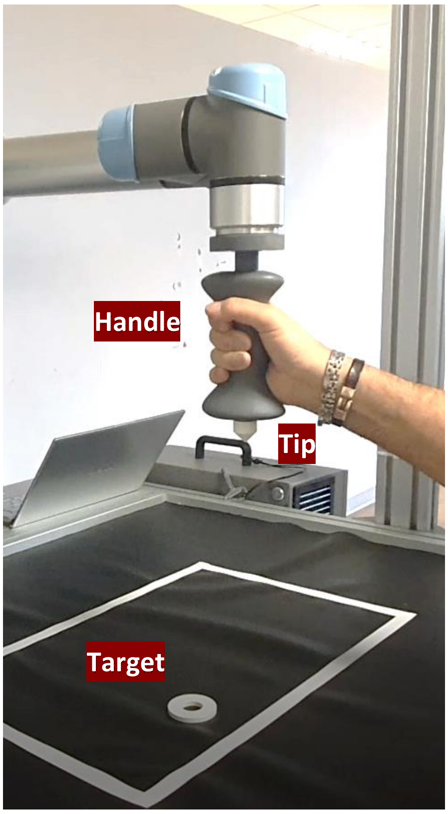


Figure 11 - Result of Handle-Target-Tip system

Figure 12 - Main component of workbench.

3.2 UR5e Calibration

The purpose of the dynamic calibration is to improve the estimation accuracy for arbitrary trajectories. Calibration process is established a transformation matrix that will convert points from machine vision space to robot space. There's really two steps to configuring vision guidance: the first step being. calibration, the second step being pattern recognition and communications. In other words. one job is going to be dedicated to calibration, and that's going to be the process of the hand eye movements of the robot moving in front of the camera's field of view, second job is sending those coordinates to create a file that translates 2D camera coordinates to six-dimensional robot coordinates.

It is requested as equipment:

- UR collaborative Robots (for this experiment was used UR5e)
- Cognex Insight 7802 Camera
- Ethernet switch: provides communication between Camera and UR, so they are just on the same network, passing data back and forth over Ethernet.

It is request as software:

- Polyscope 5.8
- Insight Explorer 5.9
- Cognex vision guidance URCapabilities (URCap): they are certified drivers from third party vendors that allow easier configuration of complex applications. In this case is going to be vision guidance.

It is request as theoretical concepts:

- Hand-eye calibration refers to the process where the robot is moving throughout the field of view automatically and the camera is finding the robot
- Pose transformation: the camera sees all its objects in a 2D pixel coordinate system, then a pose transformation is applied. These coordinates are translating into six-dimensional robot

coordinates. and then finally to get those coordinates in we are going to be using Ethernet communication.

- Ethernet communication: acquire these six-dimensional coordinates and provide the correct communication between camera and UR5e

Figure 13 - Required (and additional) components to perform a calibration process.

Here below, there are the main steps to perform a calibration process:

1. Install all the required software in the PC and Teach Pendant (robot interface)
2. Connect Camera System and Robot: they need to work on the same network (same IP addresses). Once it is established a communication over the ethernet connexions it is possible to start programming.
3. Adjusted exposure of the Camera System
4. Define a correct TCP of the robot
5. Locate the robot in the field of view defined by the camera system.
6. Start the 'Calibration'command. This process requires to set three points: two points to define the width and one point that defines the height, as shown in Figure 14, just above.

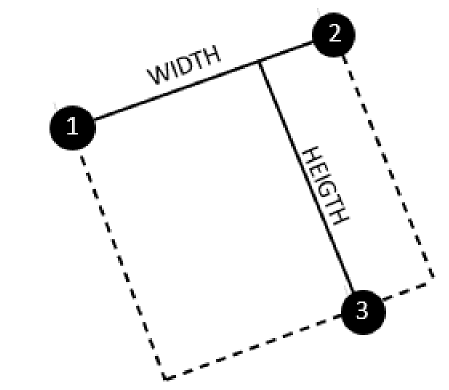


Figure 14 - Setting of three points (1,2,and 3) to perform the calibration process. They define the width and the height of the field of view with the respect to the camera.

7. Robot is now starting to move along these positions and automatically generating the calibration file. There are several different positions that the robot is generating based off the field of view coordinates that would set before. For this process, a specific tip (Figure 15) was implemented as part of the robot system. The TCP point is set at the top of the tip. The tip measures 8cm in length and has a radius of 1,5cm; this specific shape was chosen to allow the user to visualize the different positions during the calibration process. This tip is only used during calibration process, tip (represented in Figure 9) is instead applied every time the subjects are running the exercises.

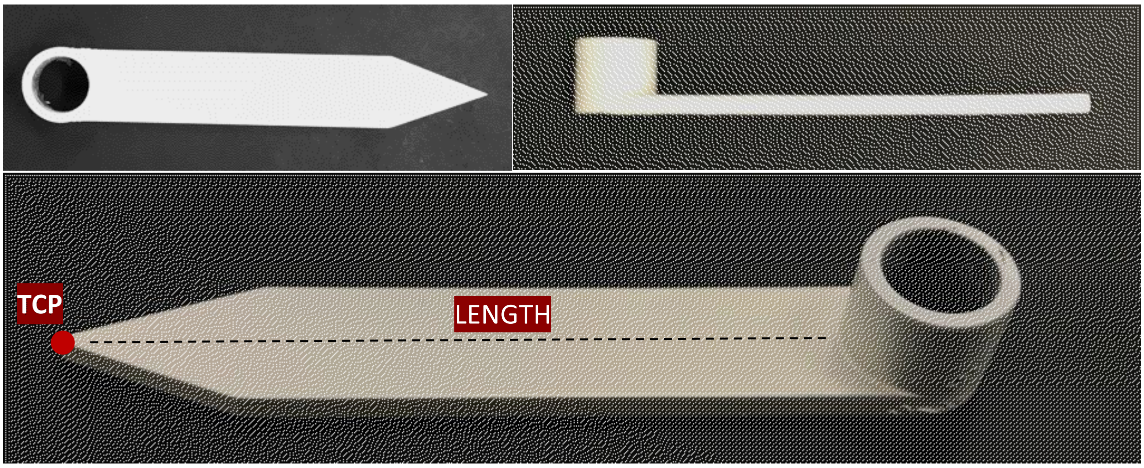


Figure 15 - Tip shape. The red spot at the top of the tip is pointing the TCP.

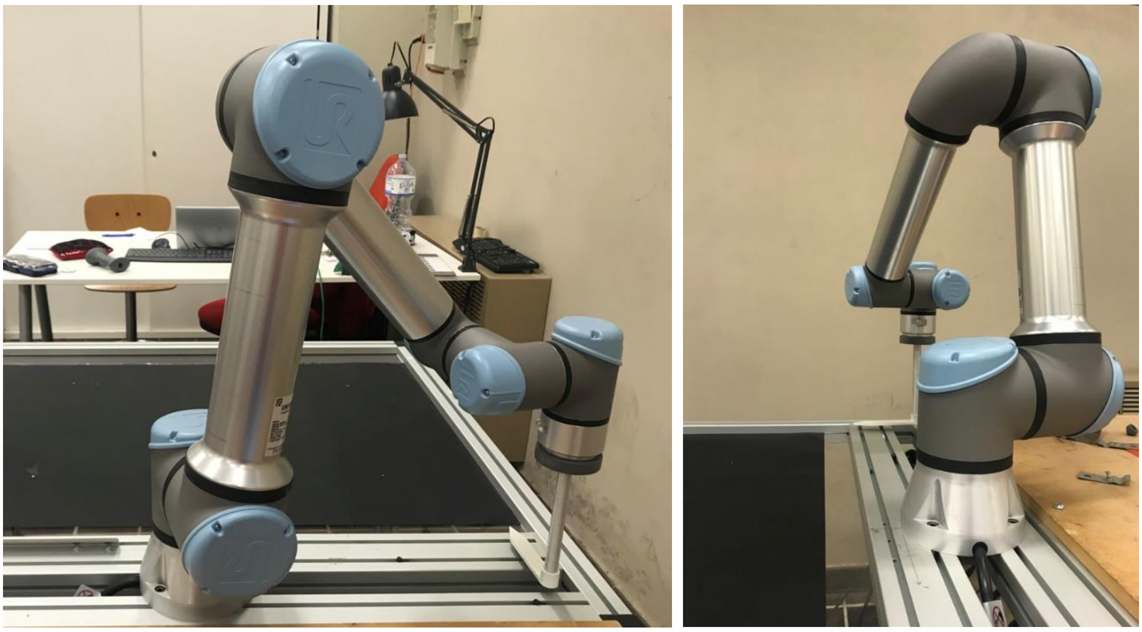


Figure 16 - Tip integrated in the Robot system.

3.3 Definition of the rehabilitation exercise

Since calibration process is performed, it is possible to carry out the rehabilitation exercise. In the proposed case, the patients are asked to sit down and relax. The aim is to bring the robot's tip to the target identified by the camera and carry out this exercise at least three times using both sides, the healthy and affected ones. Since this experiment is for investigative and preliminary purposes, many repetitive cycles are not necessary as literatures claim. The main objective is not to tire patients but to understand if this type of system is feasible in large numbers. Once the cycle is completed in F-Modality, it will be required to perform the same exercise also in other modes (M and D- Modalities).

In the proposed case, the intervention of an operator is required to set the working parameters. The program is it composed by many steps:

1st: the target is carefully positioned in the working area.

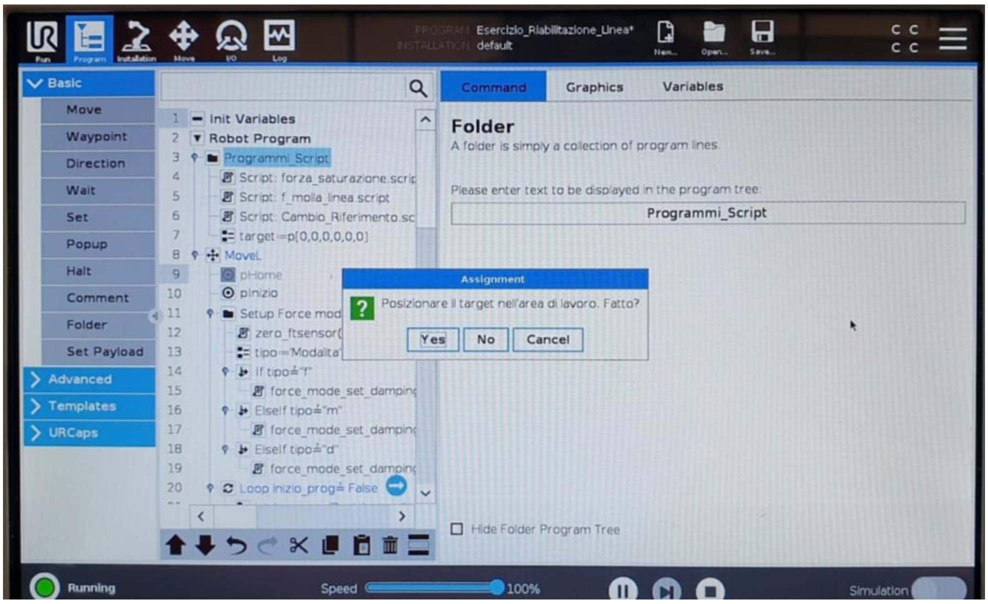


Figure 17 - The target is carefully positioned in the working area.

2nd: on the Teach Pendant, the operator is asked to select the desired exercise mode, or the degree of difficulty to apply the patient. There will be shown three modality of exercise that determines the difficulty of the exercise. Talking with engineering words, this difficulty is depending on one parameter proper of the robot: the damping coefficient. This value can be setting in a range from 0 to 1 and the more difficult the exercise, the closer it will be to value 1.

F-Modality (easy mode): the operator enters the letter "f" and clicks Submit. Easy mode is the lightest mode for the patient and the robot is very fluid in movement.

M-Modality (medium mode): the operator enters the letter "m" and clicks Submit. The patient applies little force to allow movement.

D-Modality (difficult mode): the operator enters the letter "d" and clicks on Submit. The patient applies a high force to allow the robot to move.

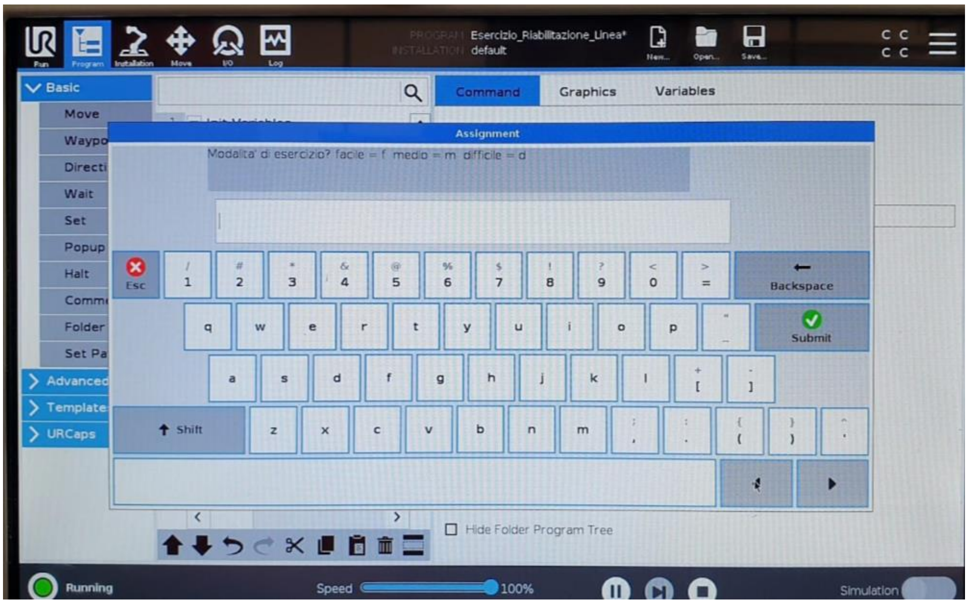


Figure 18 -. Modality of the exercise: easy, medium, or difficult, visualized in the robot interface.

3rd step: establish the number of repetitions of the exercise. The operator sets the number of repetitions of the exercise (number of cycles) for each modality. For patients was chosen to repeat the exercise 3 times, while for the healthy subjects was chosen 5.

- ➔ R of patients = 3 times for each modality
- ➔ R of healthy subjects = 5 times for each modality
- ➔

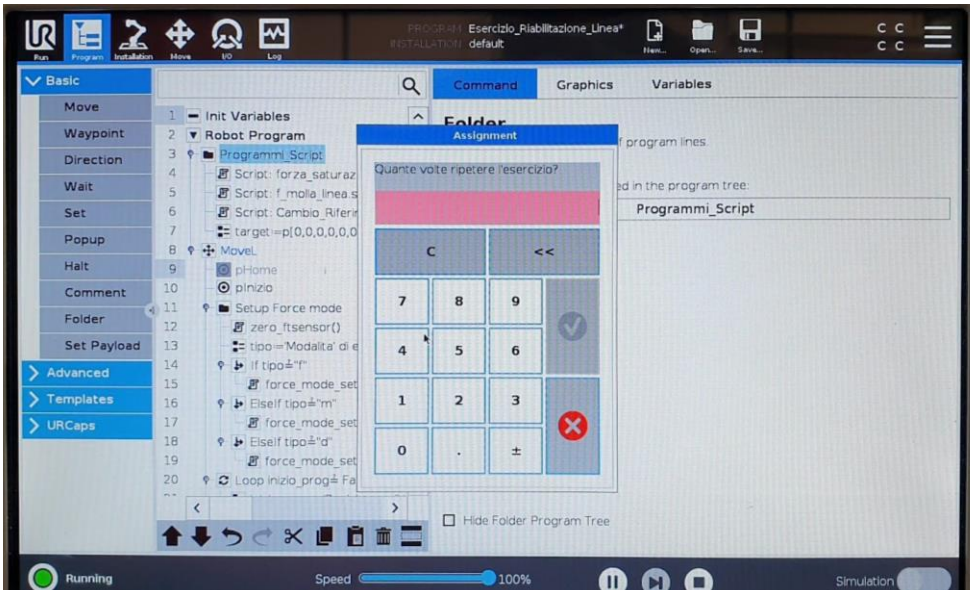


Figure 19 - The operator must establish the number of repetitions of the exercise

4th : A setting starting point is necessary to begin the exercise: this position must be comfortable for the patient, allowing movements in all direction; for that reason, the patient grips the ergonomic handle and decides where to stop, since the robot is running in free drive modality. An operator may help the patient during this process. In Figure 20 it is shown the setting of the initial starting point. This point is only valid for left side as shown in the figure; to perform the exercise with the other side is necessary to reset a starting point.

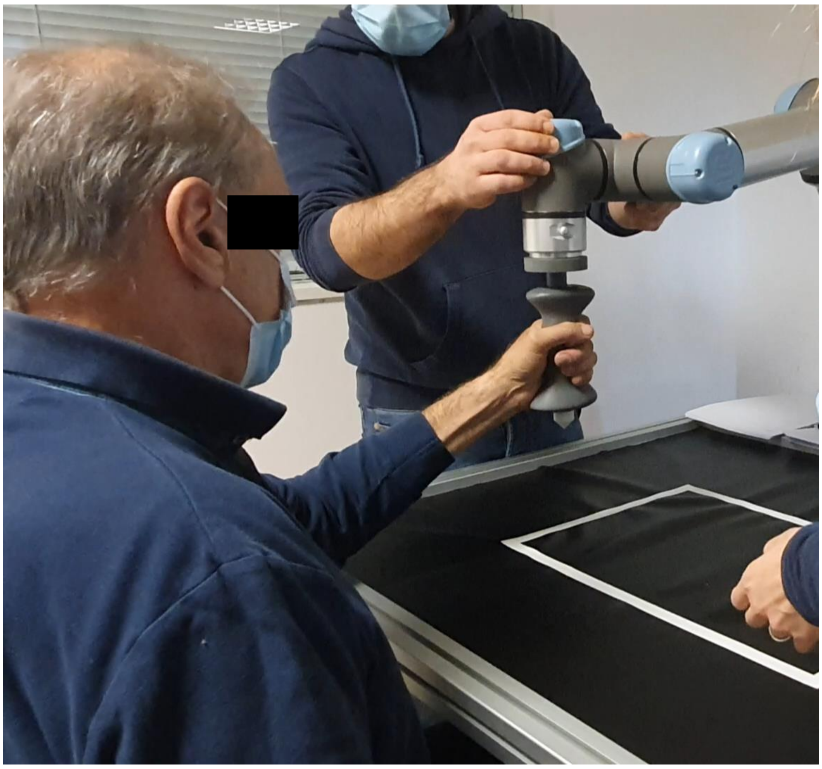


Figure 20 - Setting the starting point. An operator is helping the patient to feel as much as comfortable.

5th: from the starting point, a measurements step is now required. By gripping the handle, it is asked to the patient to relax as much as possible to weigh the upper limb. Moreover, it is asked to apply a “maximum” force (depending on the limit of the robot system) in one direction chosen by the patient. Most of them, chose to push towards.

6th: The patient is ready for the exercise. If the patient deviates from the defined trajectory, the robot will oppose with a force proportional to the distance travelled and will bring the patient back to the preferred trajectory. Once the target is reached, the robot comes back at the starting position and patient can start again.

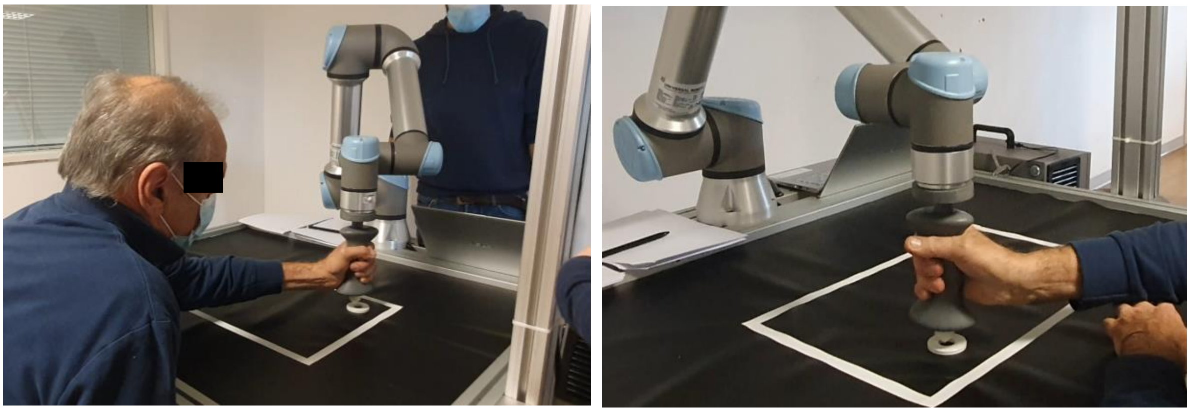


Figure 21 - The patient is performing the exercise and the target is correctly reached.

3.4 Language of the Robot: Scripts

The entire script, that is reported in Appendix A, is composed by three main threads (programs that run in parallel during the exercise):

1st thread: here are declared all the variables required to start the program. Every variable declared inside a program has a scope, that is the textual region where the variable is directly available. Two qualifiers are offered to modify this visibility:

- local qualifier informs the controller to treat a variable inside a function, as being truly local, even if a global variable with the same name exists.
- global qualifier forces a variable declared inside a function, to be globally accessible.

In case no local or global qualifier is specified (also called a free variable), the controller will first try to find the variable in the global and otherwise the variable will be treated as local [10].

Moreover, it is defined the connection between the camera and the robot interface by Cognex In-Sight Robot Guidance, the driver mentioned above.

2nd: in this thread, two parametric data are required to approximately measure: the weight of the arm slightly clinging to the ergonomic handle (or better "dead weight") and the maximum force that the subjects can apply according to a preferred direction (most of the patients chose to push towards). The first parameter is required to calculate K , while the second parameter (called 'F_vera') is required to calculate the external forces exerted at the TCP. The force 'F_vera' is the norm of Fx, Fy, and Fz. The three modalities (easy, medium, and difficult one) are strictly depending on this value, since:

- in easy modality (F-Modality) the robot opposes a force that is 20% of 'F_vera';
- in medium modality (M-Modality) the robot opposes a force that is 50% of 'F_vera';
- in difficult modality (D-Modality) the robot opposes a force that is 90% of 'F_vera';

In this first thread, f_molla' is calculated, since strictly depends on the distance between the TCP and the target position. There's only one allowed trajectory to arrive at the target and this is a linear trajectory. For that reason, it is introduced 'f_molla', a parameter that resists if the patient goes out of the predefined trajectory; in this perspective it helps patients to arrive at the target position by following the only and corrected trajectory.

3rd thread: here the proper exercise can now start. It is requested:

- new start position of the robot, from which the patient will begin to perform the exercise, defined as:

```
#pstart = punto iniziale
    x_i = pstart[0]
    y_i = pstart[1]
    z_i = pstart[2]
```

- target position in the working area, defined as:

```
#target = punto finale
 x_f = target[0]
 y_f = target[1]
 z_f = target[2]
```

- trajectory, that is defined by a mathematic formula here reported:

```
    #vettore s

    s_x = (x_f-x_i)*(1/sqrt(pow(x_f-x_i,2)+pow(y_f-y_i,2)+pow(z_f-
z_i,2)))
    s_y = (y_f-y_i)*(1/sqrt(pow(x_f-x_i,2)+pow(y_f-y_i,2)+pow(z_f-
z_i,2)))
    s_z = (z_f-z_i)*(1/sqrt(pow(x_f-x_i,2)+pow(y_f-y_i,2)+pow(z_f-
z_i,2)))
```

- choice of the upper limb to carry out the exercise (left upper side or right upper side)

- choice of the complexity of the exercise: easy, medium, or difficult

- choice of the numbers of repetitions (numbers of cycles).

The patient can now perform the exercise for as many cycles as were set. Once the tip is close to the target position (should measures less than 1cm), a sound is emitted, and the robot come back to the starting point to repeat the exercise. Tip proximity to the target is defined by this formula:

```
global diff=point_dist(posa_tcp_ini,target)
```

It calculates the difference between the initial position of the TCP and the target position.

CHAPTER 4

CLINICAL EXPERIMENTS

In this chapter, it will be present the main data of the patients and the results will be analysed. Subjects were volunteered in the neuro-motor rehabilitation department of Torrette Hospital, Ancona. Moreover, it will be present the clinical results of pathological patients and of the healthy ones. The data were extrapolated by UR Log Viewer for position. This software returns information about three main parameters: the change in position, the change in velocity and the change in acceleration. It was also implemented a Matlab code to post-process forces data. Results of pathological Patient 1 were discarded a priori.

4.1 Partecipants

The study was a randomized controlled study performed at Torrette Hospital, Ancona (Italy) including 19 participants, among which 10 were chronic subjects (mean age: 67- and 1-11-years post-disease) with diminished movement and tactile perception in their affected part. Since this investigation is a preliminary study, no prior data are available for optimal sample size calculation. The participants numbers were assigned only to present data in this manuscript. Eligible participants were randomly assigned to the UR5e robot. Before starting the exercise, everyone was asked to complete a questionnaire form (Appendix B) and have been informed of the progress of the exercise. Results of that questionnaire will be reported in the next paragraph.

4.2 Clinical data of partecipants

Patient Number	Age	Gender	Height [m]	Weight [Kg]	Side Affected	Notes
1	72	M	1,83	85	Right	
2	65	M	1,82	93	Left	
3	64	M	1,8	83	Left	
4	76	M	1,62	73	Left	
5	62	F	1,68	70	Left	
6	69	M	1,75	113	Left	
7	51	M	1,77	100	Left	
8	71	M	1,73	80	Left	Preferred contact with clinical operators
9	69	M	1,78	78	Right	
10	70	M	1,8	70	Left	Most serious patient
11					-	
12					-	
13					-	
14					-	
15					-	
16					-	
17					-	
18					-	
19					-	

Table 2 – Demographics and notes for participants in the study. The experimental group includes participants 1-10 and sham control group includes participants 11-20. The participants numbers were assigned only to present data in this manuscript.

RESULTS

4.3 Log Viewer

The software UR Log Viewer was created for reading and viewing the Support File from the Universal Robot's cobots, which are produced automatically inside each robot, and contain the log files, programs and flight reports. The software is a support tool for the user for understanding the robot behavior and have data analysis, as well to do improvements on your application. This will open 6 windows with real-time data graphs for the selected Flight Record. Each opened window contains data for the respective joint and the data of the exact moment of the fault that has happened on the robot [11]. Just information from window 6 relative to the joint 'wrist 3' (Figure 6)-were evaluated.

For pathological patients some common features are here reported:

1st Consideration: it is possible to appreciate differences between left and right sides in Position curves. The affected one (left side) shows lines and rhythms not regular, demonstrating a higher difficulty for the patient to correctly reach the target.

2nd Consideration: in F-Modality, the amplitude recorded in 'Velocity' plots reaches peaks in between the range of 8-12°/s. In other modalities, especially in the D-Modality, the hardest one, peaks can not touch 10°/s. This means that speed decreases as much as difficulty increases. The same occurs for 'Acceleration' curves.

3rd Consideration: the period, measured in seconds, to realize the full exercise is in average around:

- 58 seconds for F-Modality
- 44 seconds for M-Modality
- 40 seconds for D-Modality

considering that the left side is affected one.

- 60 seconds for F-Modality
- 50 seconds for M-Modality
- 44 seconds for D-Modality

considering that the right side is the healthy one.

4th Consideration: in both left and right side, the D-Modality requires low time to realize the full exercise. This is probably due toa closer position of the target.

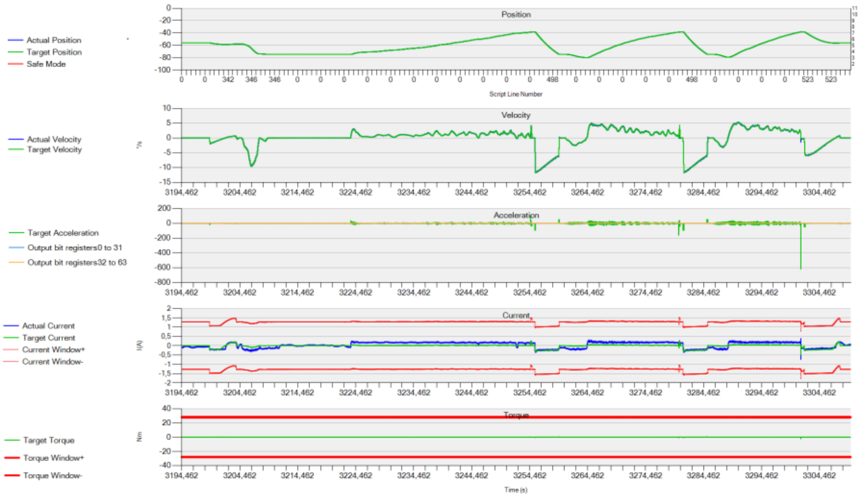


Figure 22 - Pathological patient 2. Left Side. F-Modality.

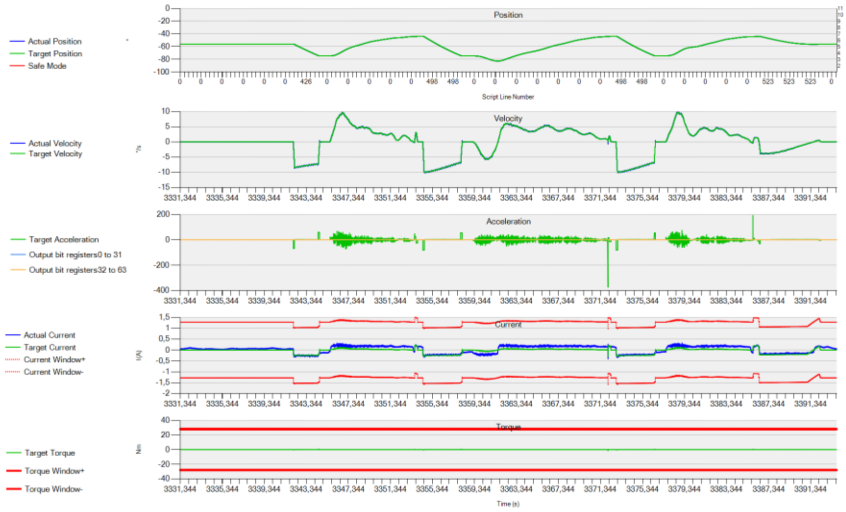


Figure 23 - Pathological patient 2. Left Side. M-Modality.

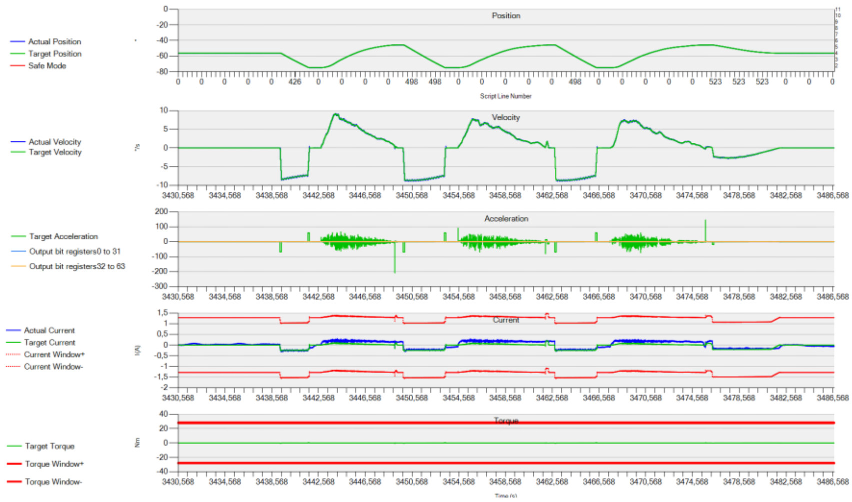


Figure 24 - Pathological patient 2. Left Side. D-Modality.

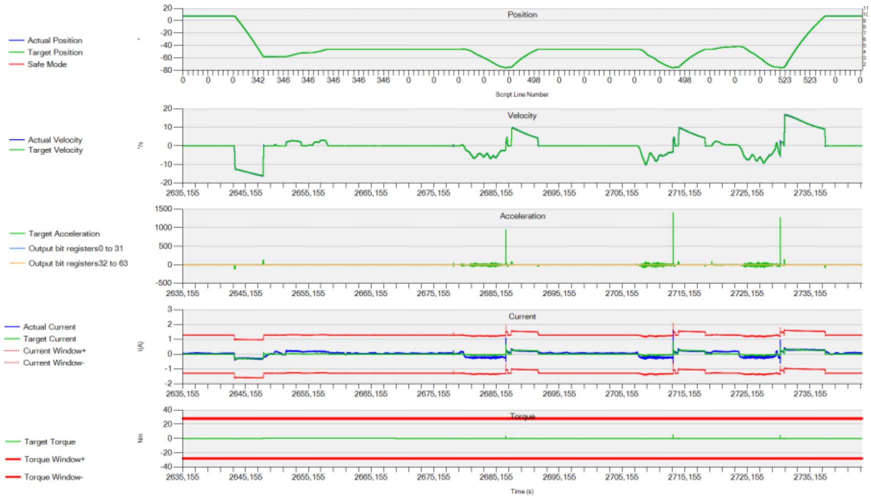


Figure 25 - Pathological patient 2. Right Side. F-Modality.

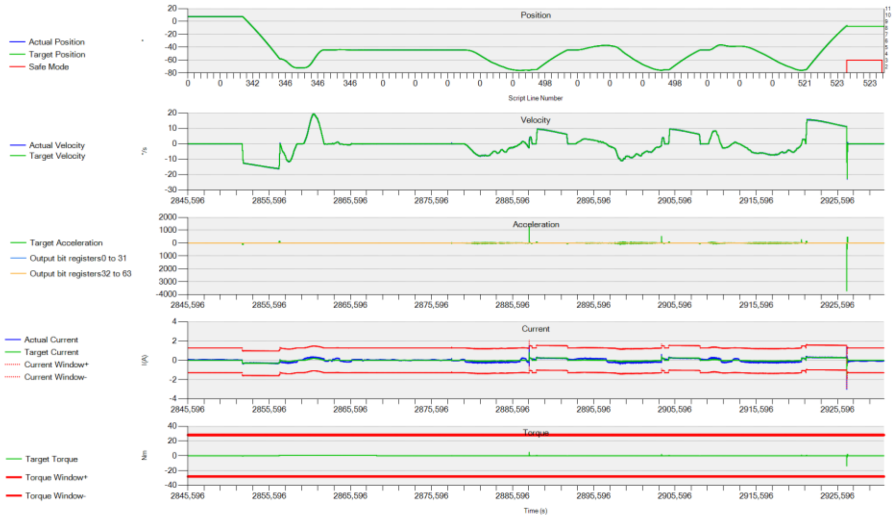


Figure 26 - Pathological patient 2. Right Side. M-Modality.

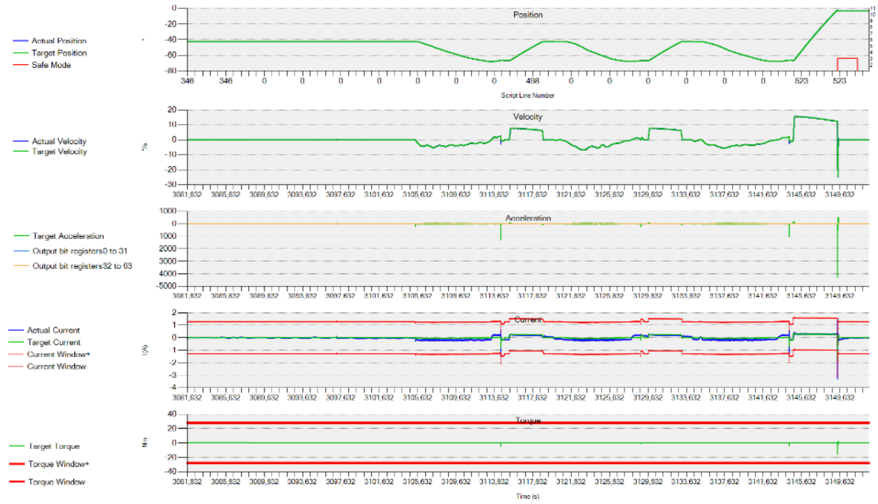


Figure 27 - Pathological patient 2. Right Side. D-Modality.

For healthy patient some common features are here reported:

1st Consideration: it is possible to point out that curves regarding ‘Position’ change in relation to the recalculated coordinates of the target. Each set of exercise is composed by 5 cycles, in which target is changing position in the working table area.

2nd Consideration: in F-Modality, the amplitude recorded of velocity reaches peaks in the range of 10-20 °/s. In the other modalities, especially in the D-Modality, the hardest one, picks touch 10°/s. This means that speed decreases as much as difficulty increases. The same occurs for ‘Acceleration’ curves.

3rd Consideration: the period, measured in seconds, to realize the full exercise is in average around:

- 42 seconds for F-Modality
- 41 seconds for M-Modality
- 45 seconds for D-Modality

considering that the left side is not for most of the healthy subjects the dominant one.

- 46 seconds for F-Modality
- 42 seconds for M-Modality
- 54 seconds for D-Modality

considering that the right side is for most of the healthy subjects the dominant one.

4th Consideration: in both left and right side, the D-Modality requires high time to realize the full exercise. This is because there is high damper coefficient executed by the robot.

Part C in the Appendix showhs all the result from the other subjects.

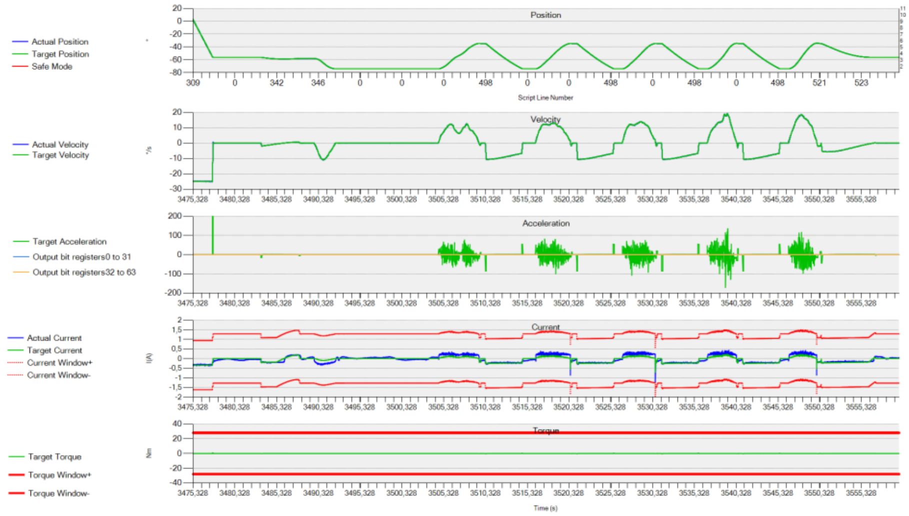


Figure 28 – Healthy subject 1. Left Side. F-Modality

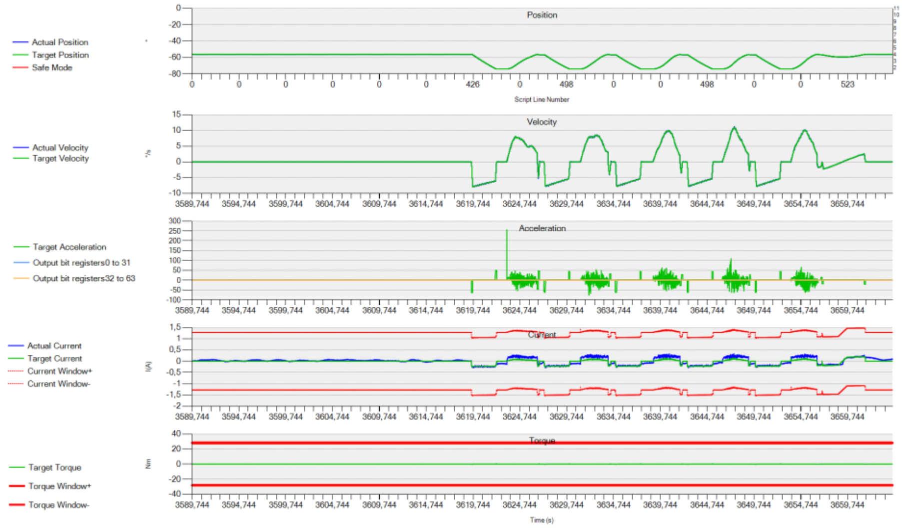


Figure 29 - Healthy subject 1. Left Side. M-Modality

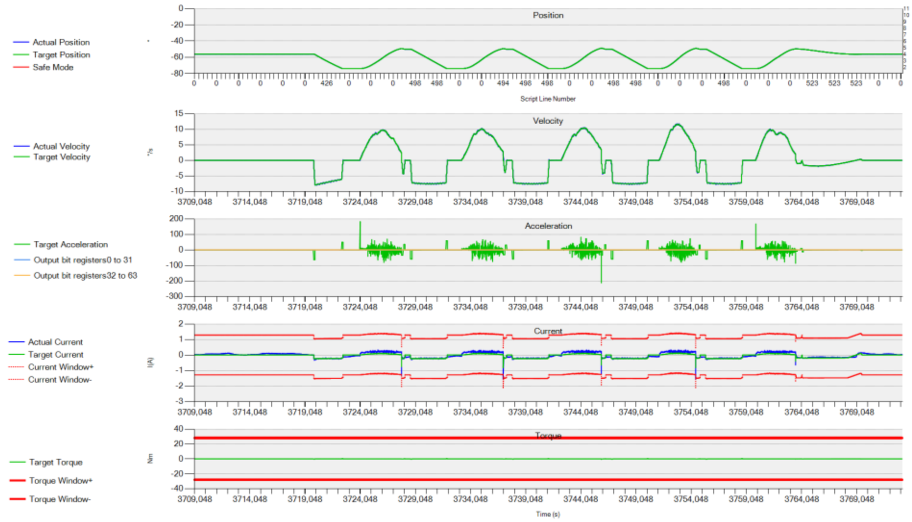


Figure 30 - Healthy subject 1. Left Side. D-Modality.

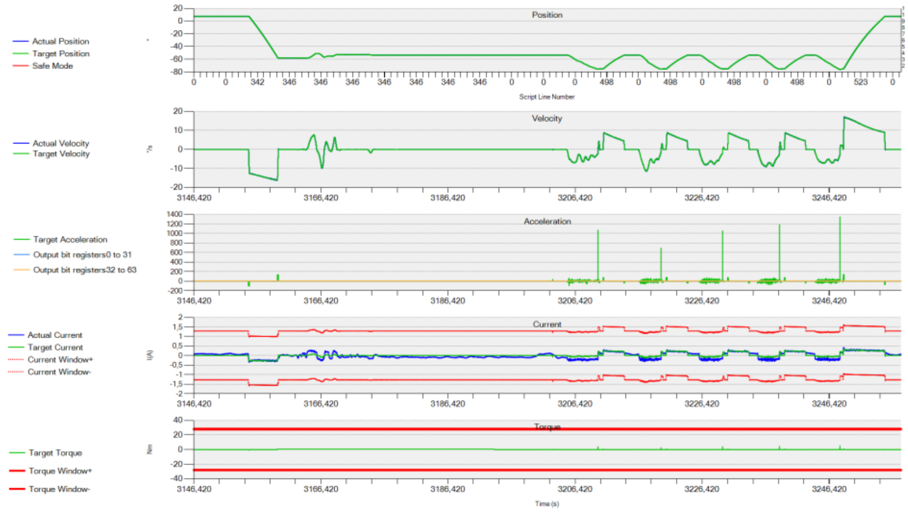


Figure 31 - Healthy subject 1. Right Side. F-Modality.

Figure 32 - Healthy subject. Right Side. M-Modality.

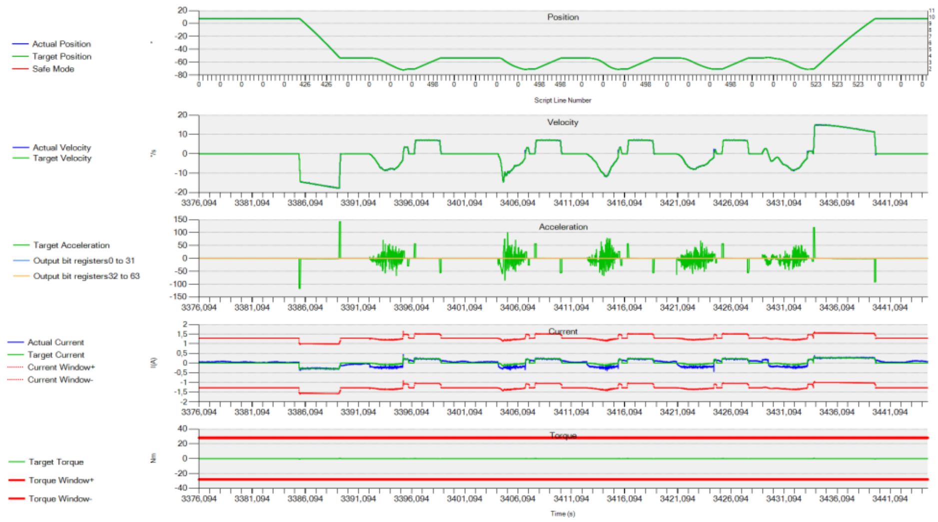


Figure 33- Healthy subject 1. Right Side. D-Modality.

4.4 Force Analysis

In Appendix Part D it is shown the script used in Matlab to visualize the analysis force.

In the graphs, it is possible to appreciate:

- The trajectories executed by the robot system
- The modulus of the radial force (any force that acts in a straight line)
- The modulus of axial force (any force that acts directly on the central axis of an object)
- The modulus of the force exerted by the robot

It is also computed in F,M and D-Modalities:

➔ F_0 as the modulus of the force applied to the TCP at time T_0

➔ F_{max} as the maximum force recorded at the TCP during all the exercises

This table here below shows the difference in the attitude between right side and left side in both pathological patient (1-10) and healthy one (11-19).

	RIGHT SIDE			LEFT SIDE		
Patient n°	*N° of repetitions*	F_0	F_{max}	*N° of repetitions*	F_0	F_{max}
1	-	-	-	-	-	-
2	3	6.5N	25.9N	3	9.2N	23.3N
3	3	5.1N	24.0N	3	17.1N	44.4N
4	3.6	2.9N	44.1N	3.3	1.5N	64N
5	-	-	-	3	5.9N	37.3
6	3.3	20.6N	94N	3	9N	46.8N
7	-	-	-	-	-	-
8	3	18.5N	83.5N	3.3	14.9N	21.9N
9	3.3	15.5N	17N	3	3.1N	5.7N
10	3	20.2N	38.2N	3	6.3N	85.5N
11	5.3	16.6N	81.2N	5	14.7N	45.9N
12	5.3	5.1N	7.7N	6.3	5.6N	10.9N
13	5.3	15.3N	40.7N	5	6.7N	50.2N
14	5	25.6N	36.5N	5	18.7N	36.9N
15	5	14.3N	30.5	5	15.8N	32.4N
16	5	10.5N	100.2N	5	10N	41.2N
17	5	20.6N	42.1N	5	23.4N	26.6N
18	5	25.4N	35N	5	25.5N	29.4N
19	5	16N	40.6N	5	15.8N	30.0N

Table 3 - Main differences between right side and left side in both pathological patient (1-10) and healthy one (11-19).

1st Consideration: in pathological patient is possible to appreciate the great difference between the 2 sides, the one affected and the healthy one. For example, these characteristics are evident for patient number 3, 4, 6, 8, 9 and 10.

2nd Consideration: in healthy patient, the difference between the 2 sides is less evident than in the pathological ones. This is because both sides are healthy and can apply the same force to development the experiment. On the other side, difference between left and right upper side must be due to predominant one. For predominant arm, movements are more precise, and less force is required, while for the other one occurs the opposite.

3rd Consideration: during the data recordings some artefacts occurred; this consequently means a lack of data and a not correct visualization of the graphs, as it is especially shown in patient number 1, 3 and 5. .

Pathological patient 2

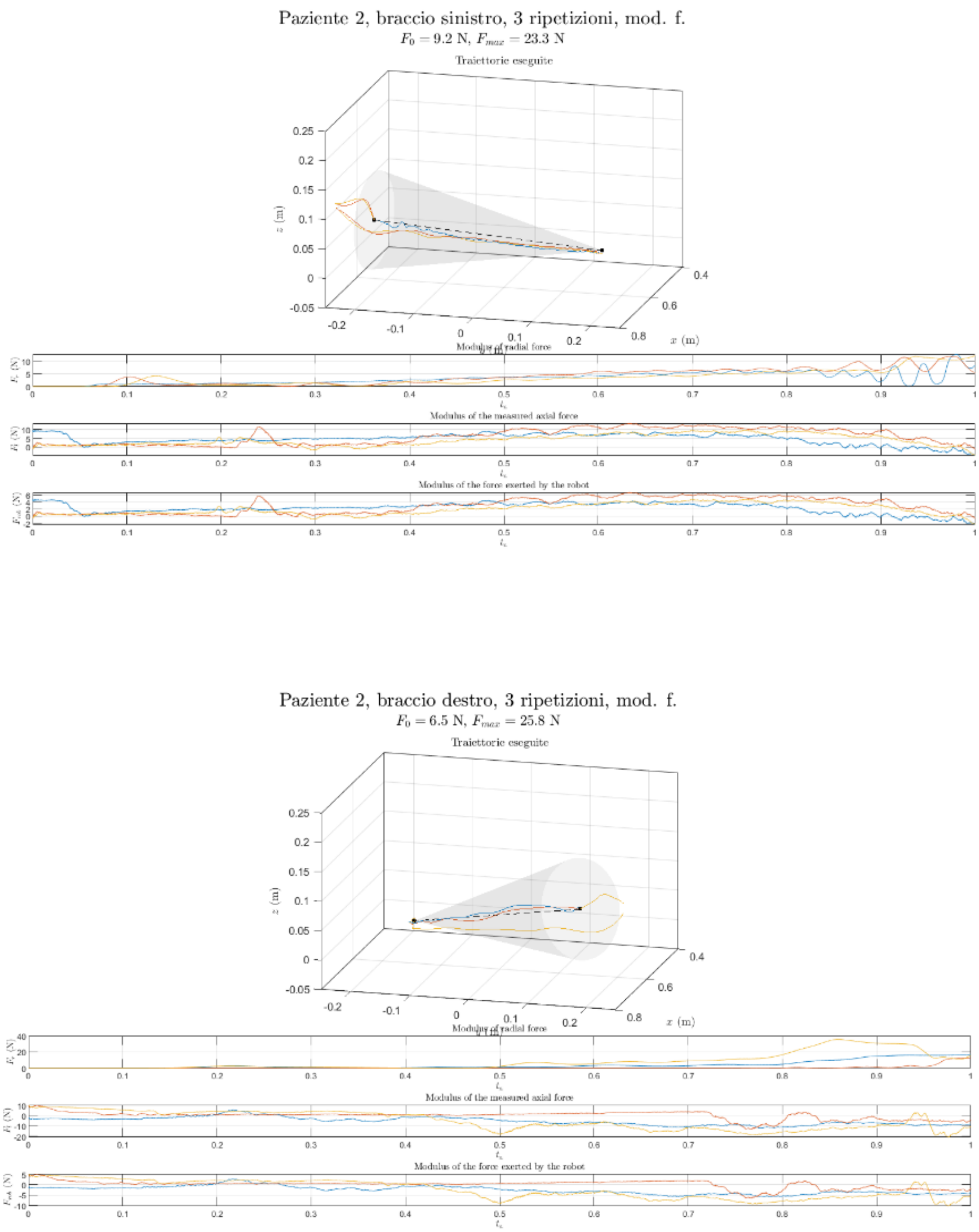


Figure 34 - Force analysis related to pathological patient 2. Right and Left side are here compared in F-Modality.

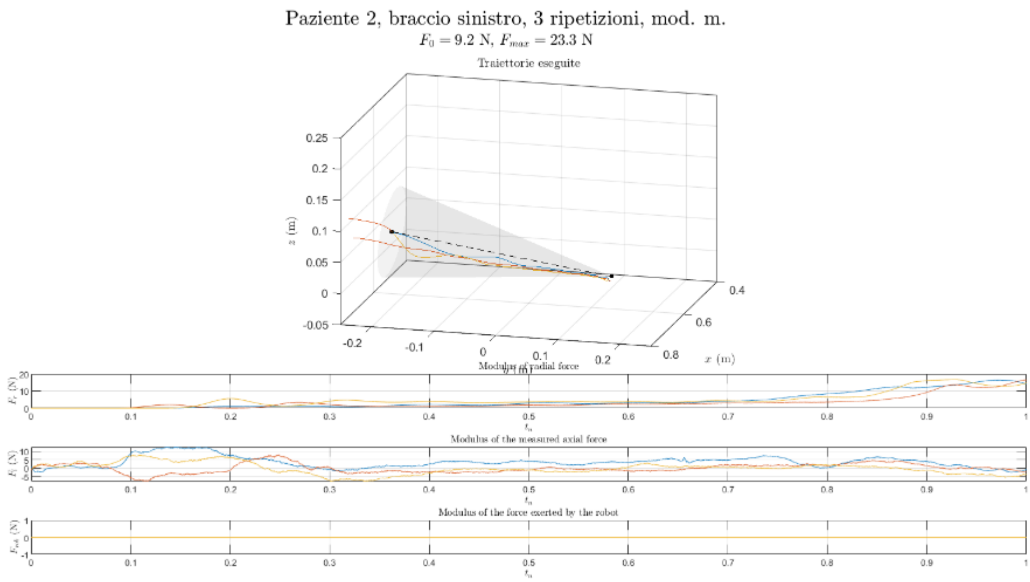


Figure 35 - Force analysis related to pathological patient 2. Right and Left side can't be compared in M-Modality since there was an error in the Matlab code.

Missing data

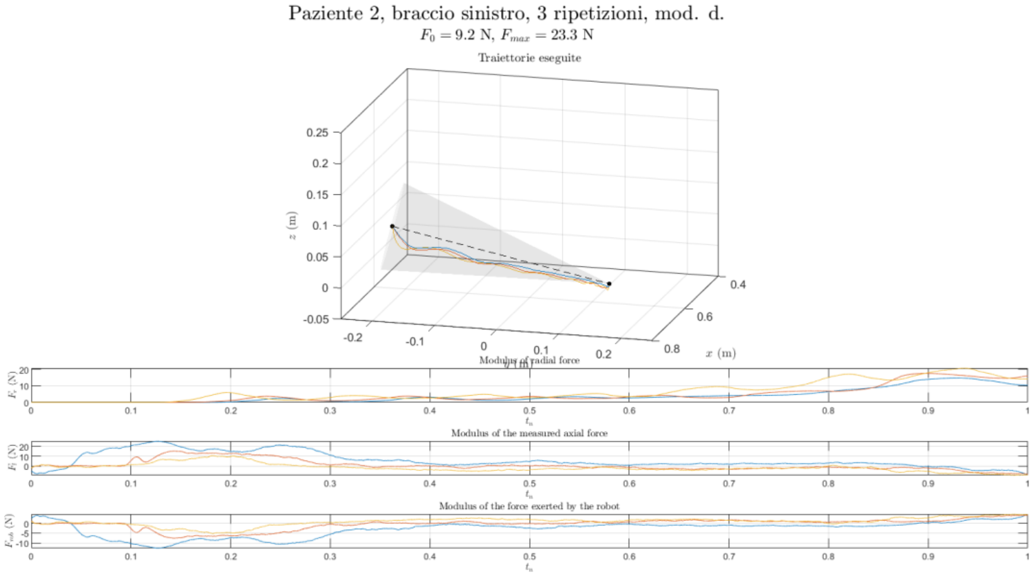


Figure 36 - Force analysis related to pathological patient 2. Right and Left side can't be compared in D-Modality since there was an error in the Matlab code.

Missing data

Pathological patient 3

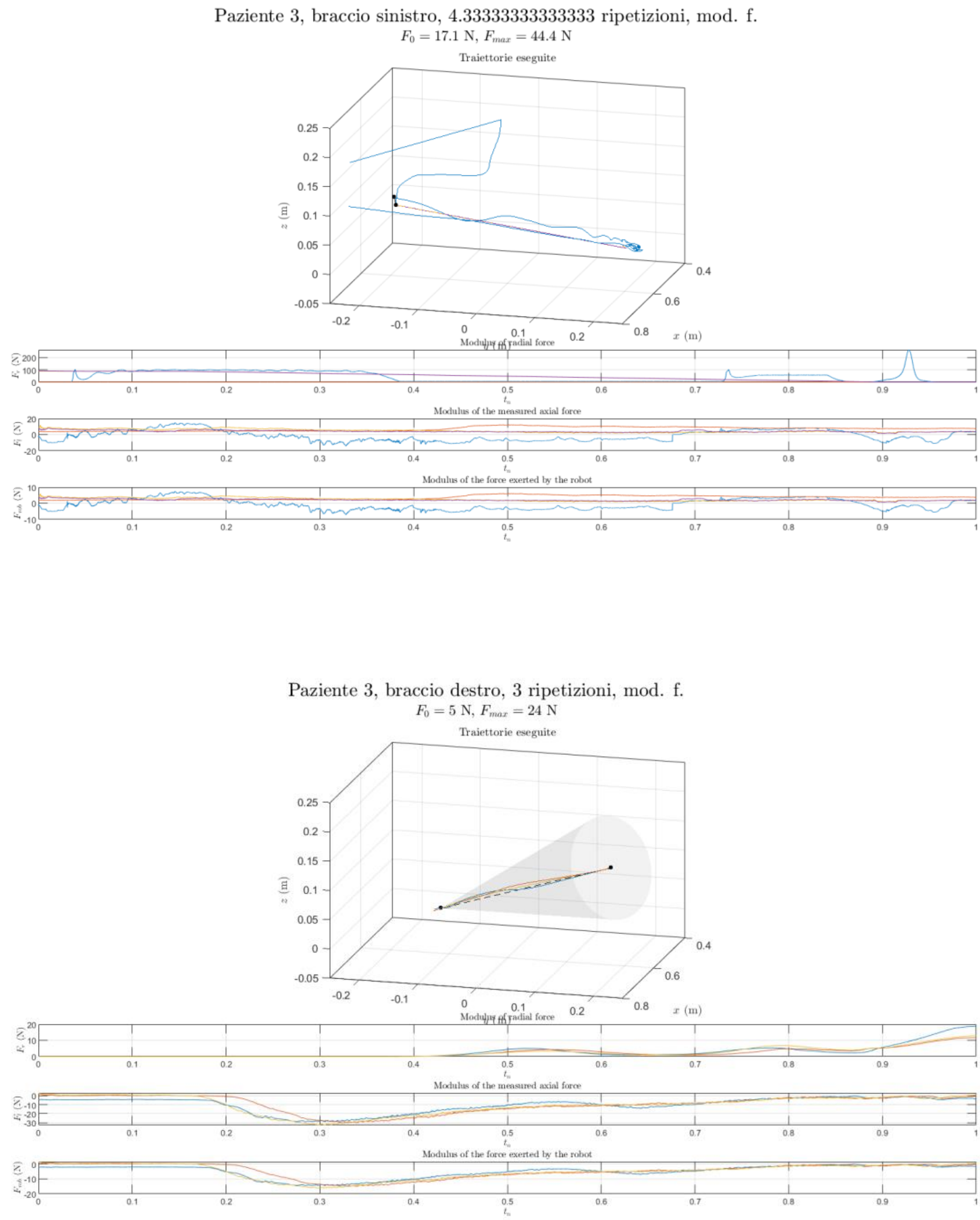


Figure 37 - Force analysis related to pathological patient 3. Right and Left side are here compared in F-Modality.

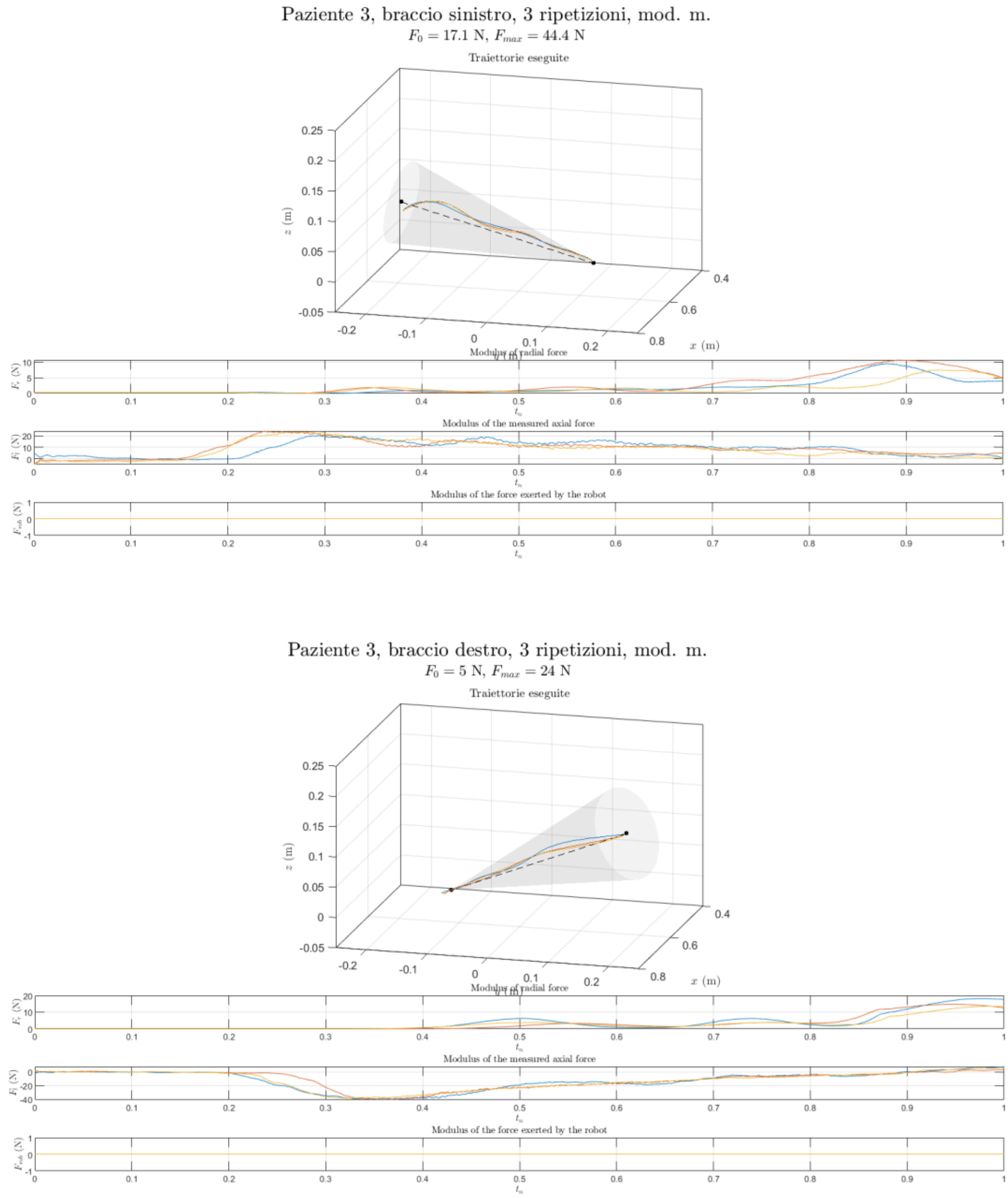


Figure 38 - Force analysis related to pathological patient 3. Right and Left side are here compared in M-Modality.

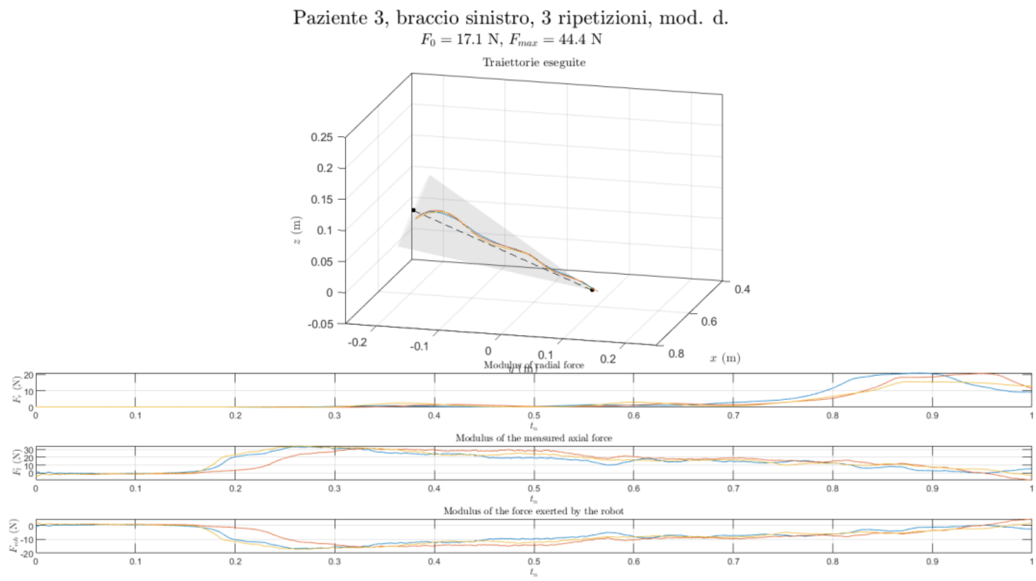


Figure 39 - Force analysis related to pathological patient 3. Right and Left side can't be compared in D-Modality.

Missing data

Pathological patient 4

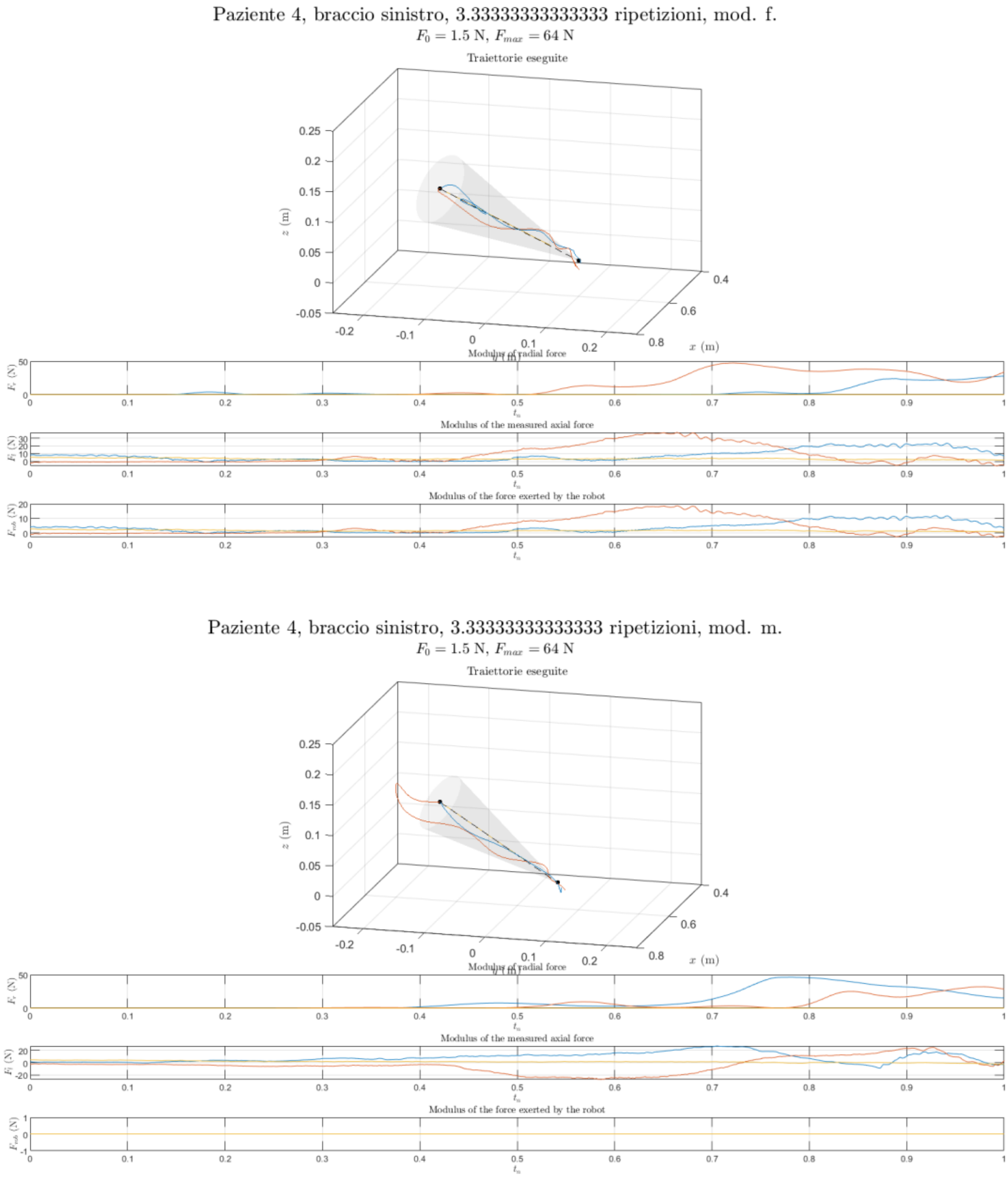
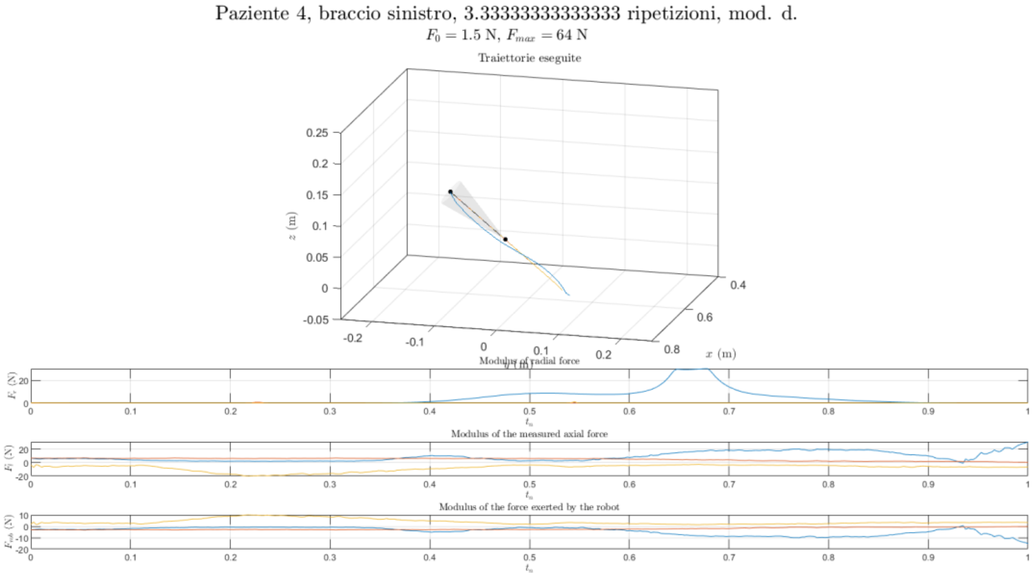


Figure 40 - Force analysis related to pathological patient 4. Left side in M-Modality.



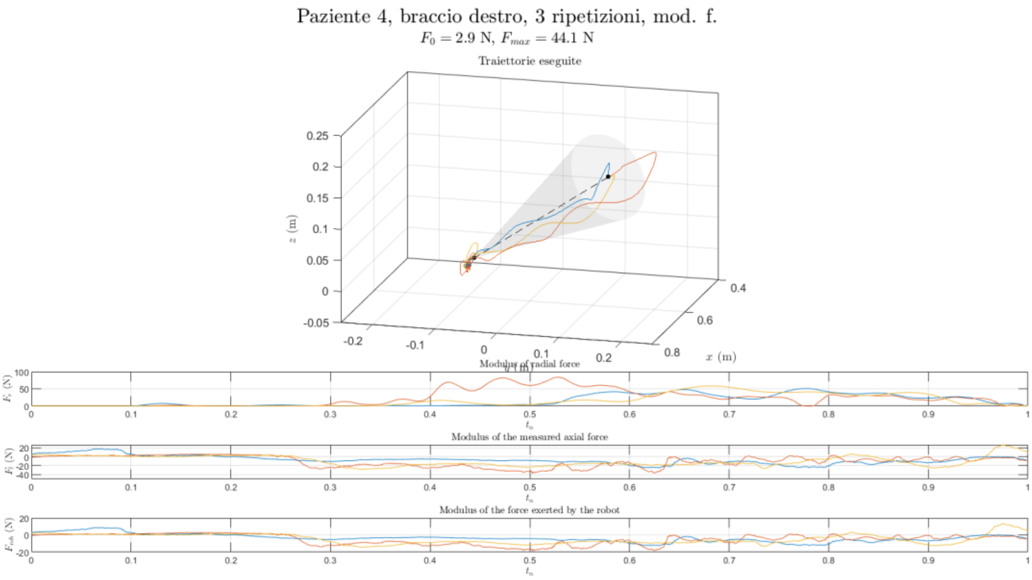
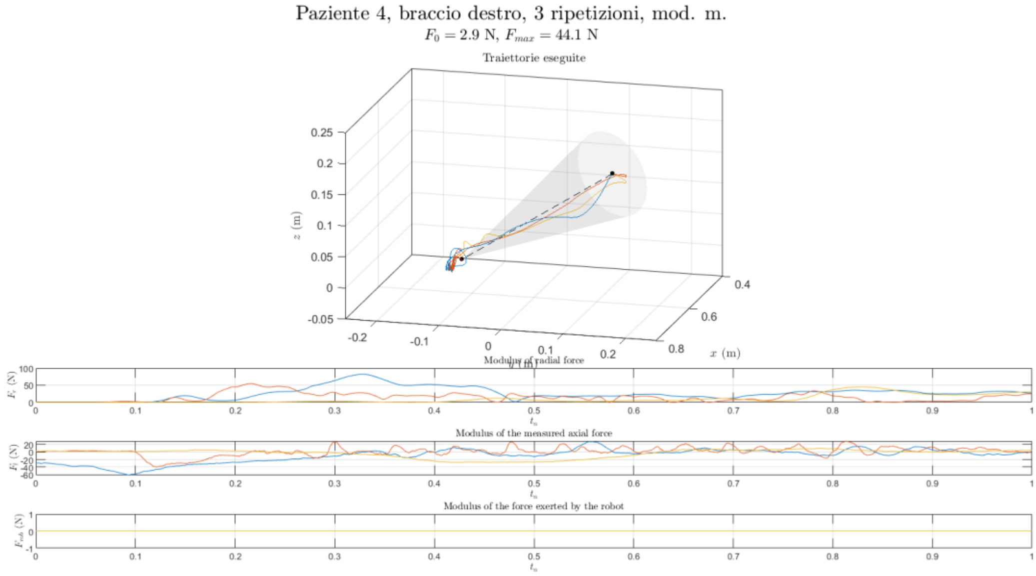


Figure 41 - Force analysis related to pathological patient 4. Lef and right side in D and M-Modality.



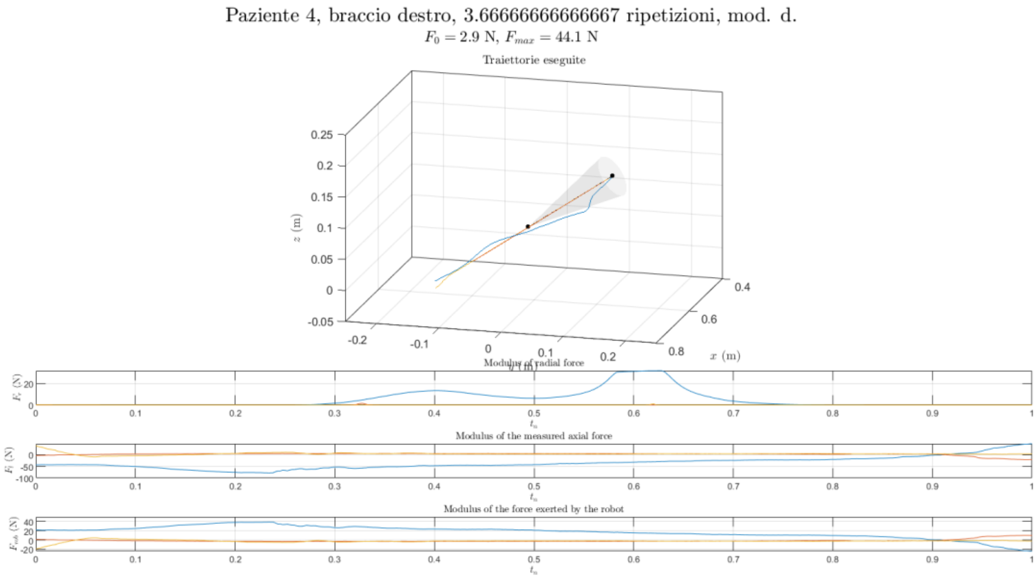
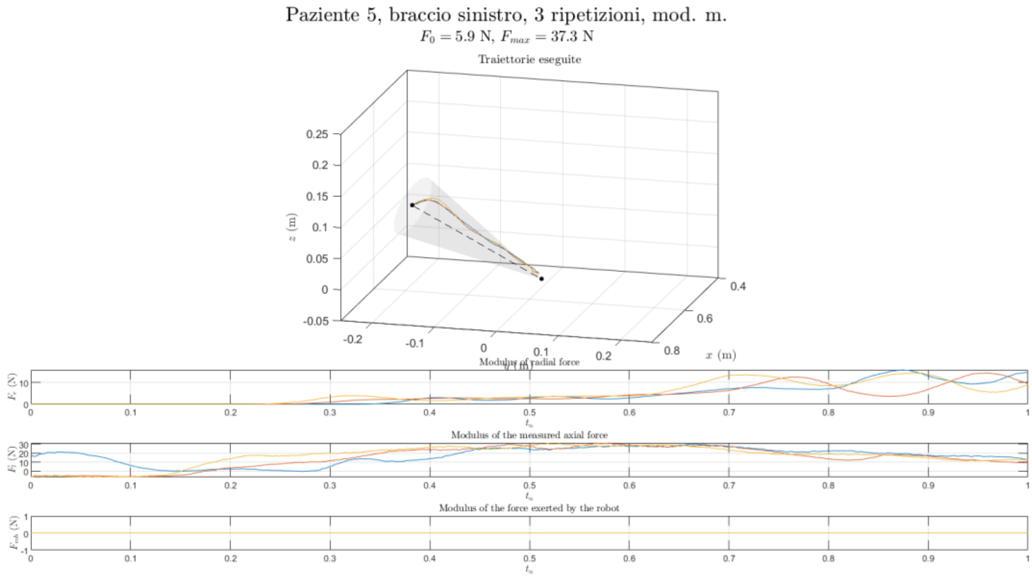


Figure 42 - Force analysis related to pathological patient 4. Lef and right side in M and D-Modality.

Pathological patient 5



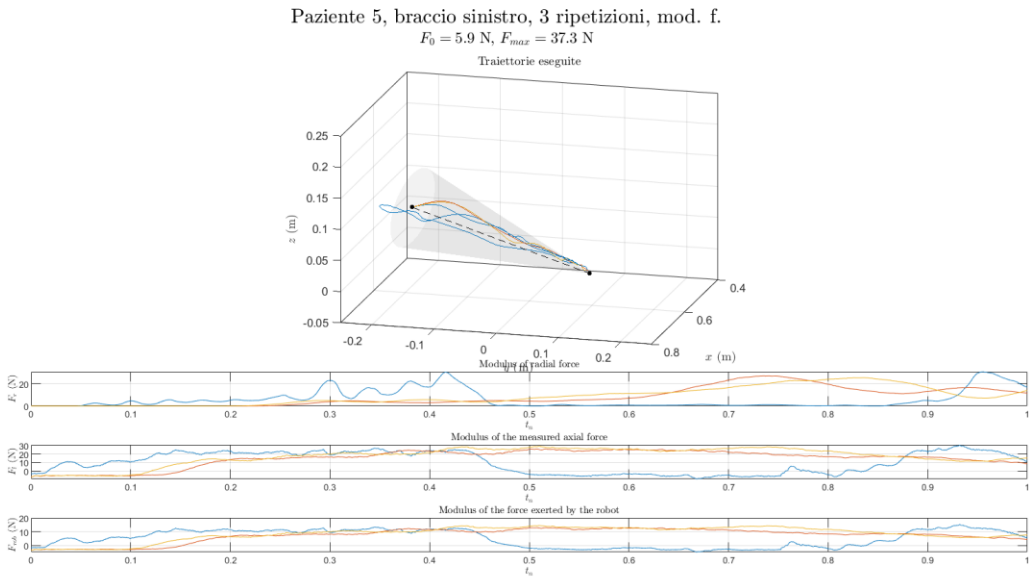


Figure 43 - Force analysis related to pathological patient 5. Lef and right side in M and D-Modality

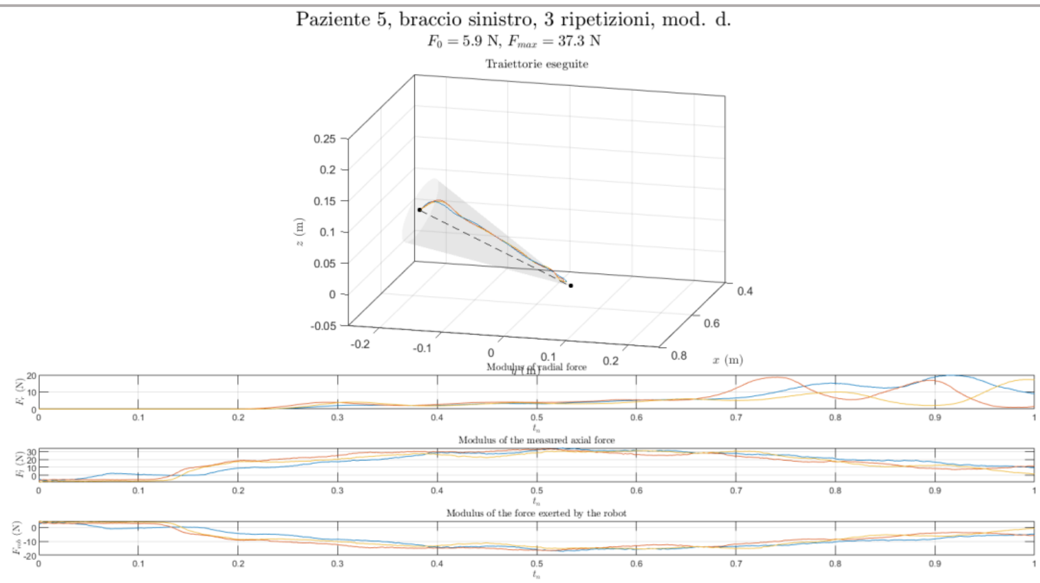
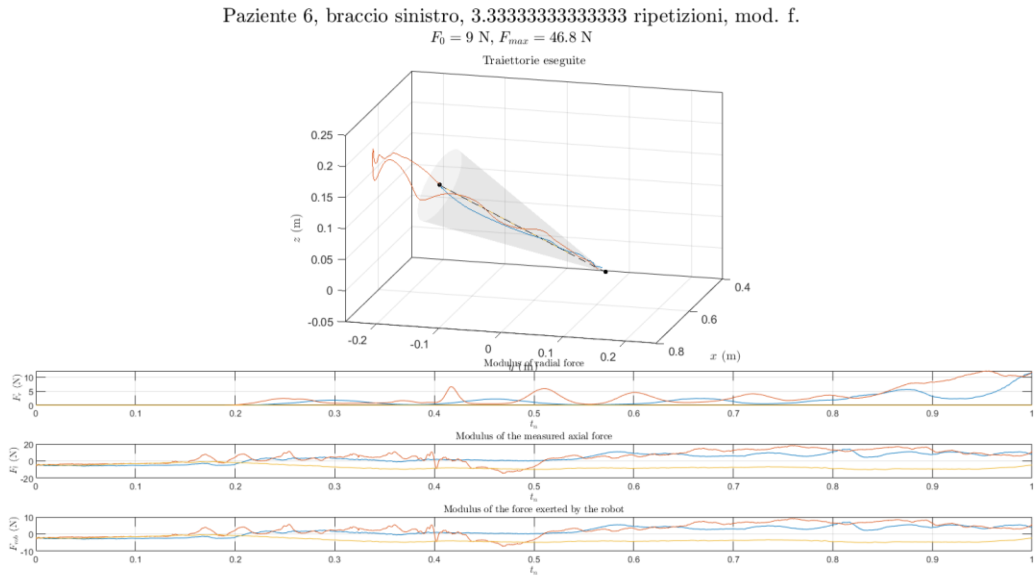


Figure 44 - Force analysis related to pathological patient 5. Lef and right side in M and D-Modality

Pathological patient 6



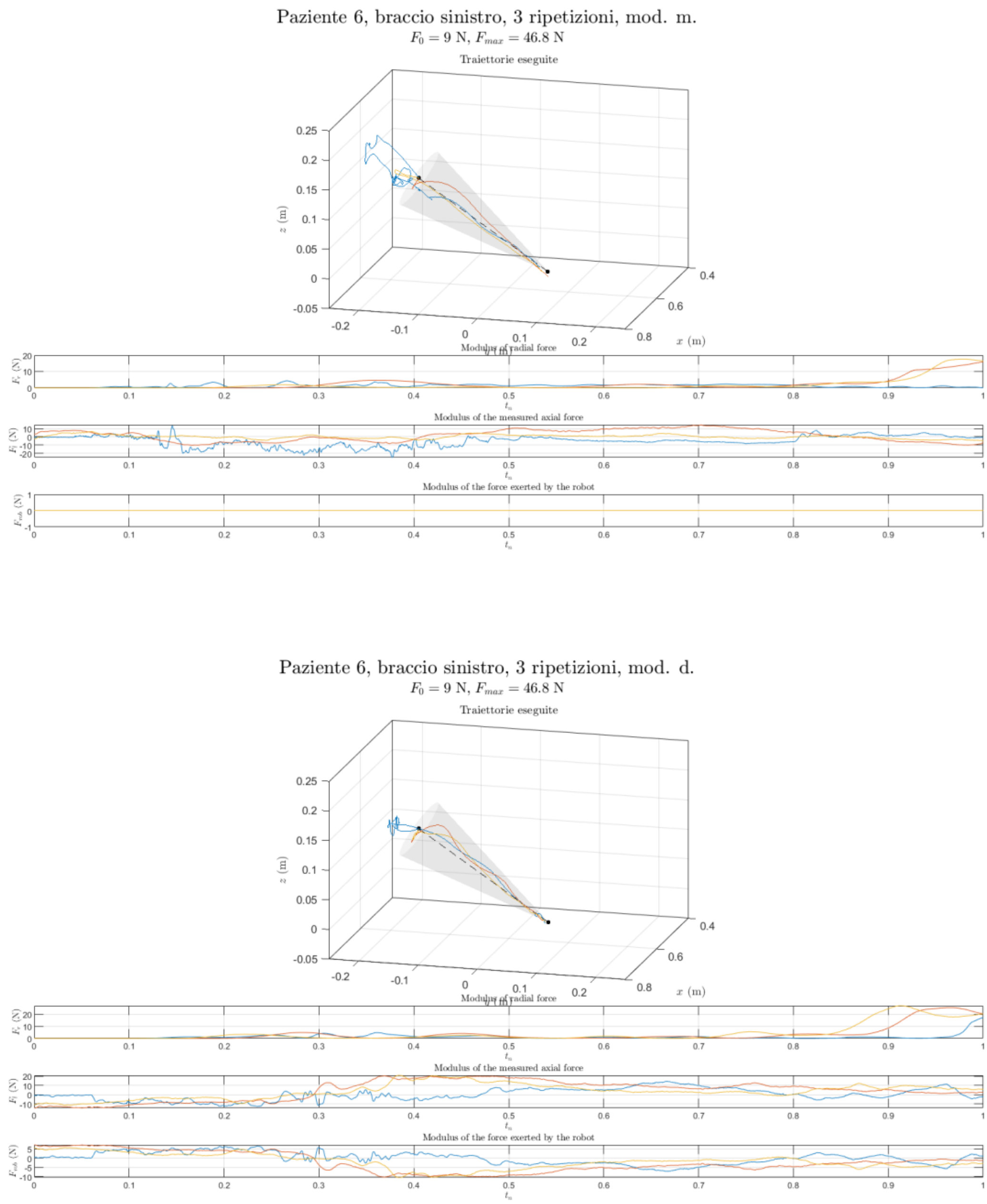
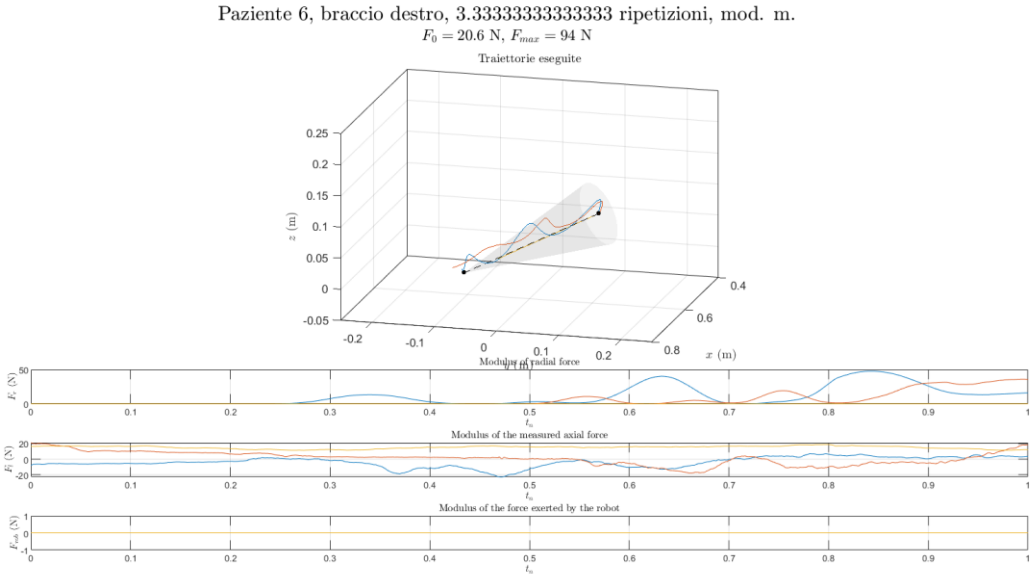


Figure 45 - Force analysis related to pathological patient 6. Lef and right side in M and D-Modality



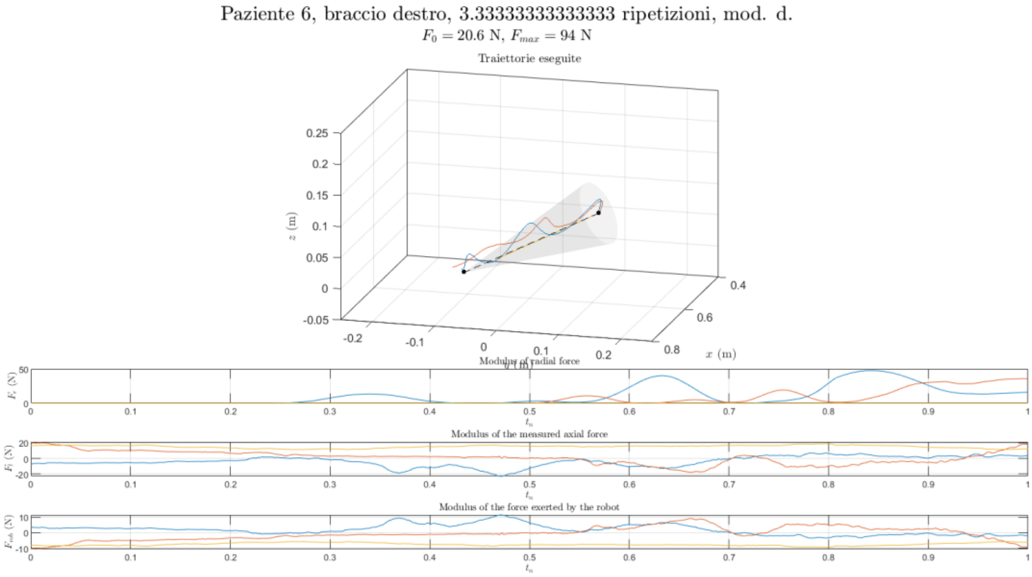
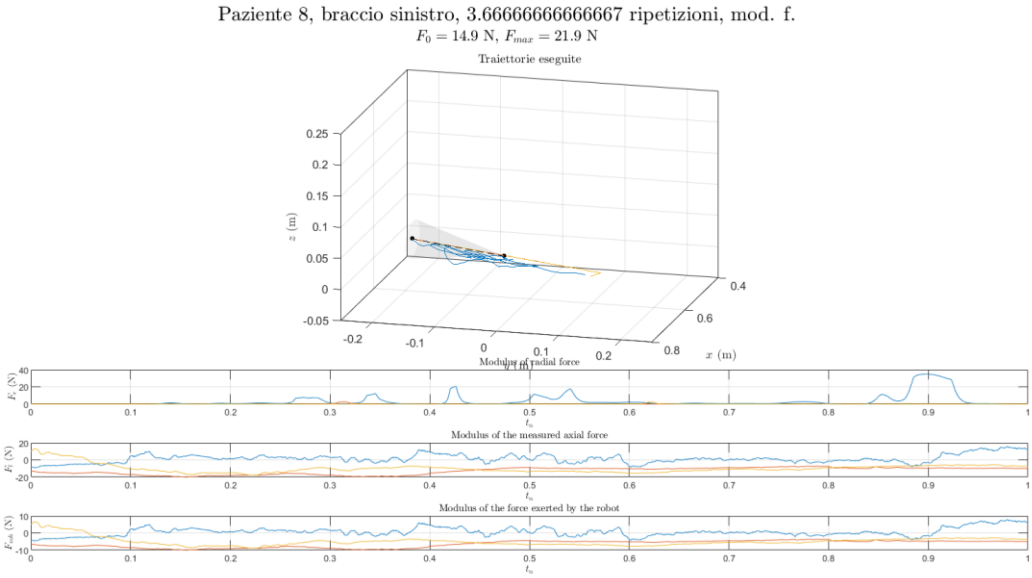


Figure 46 - Force analysis related to pathological patient 6. Lef and right side in M and D-Modality

Pathological patient 8



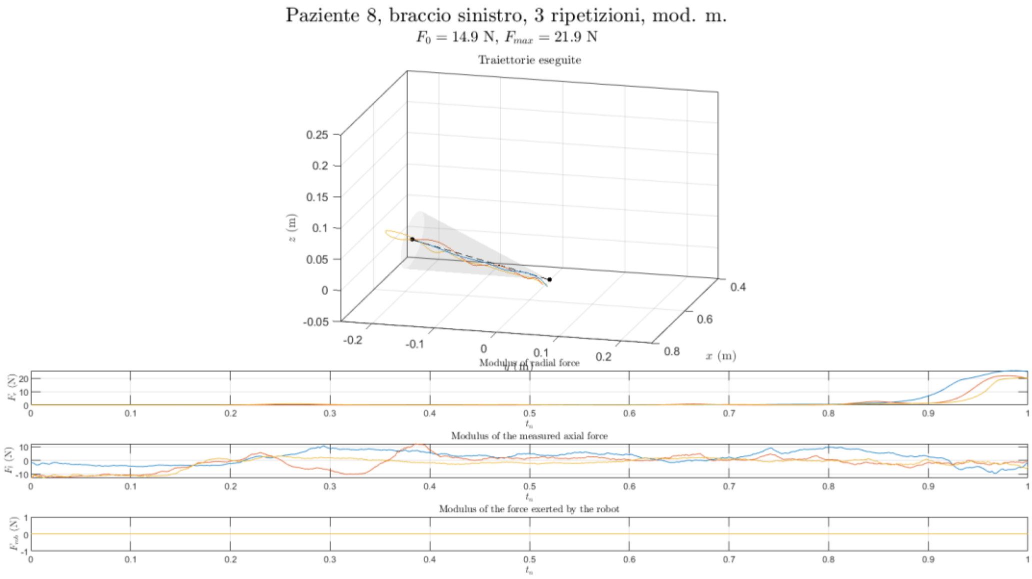
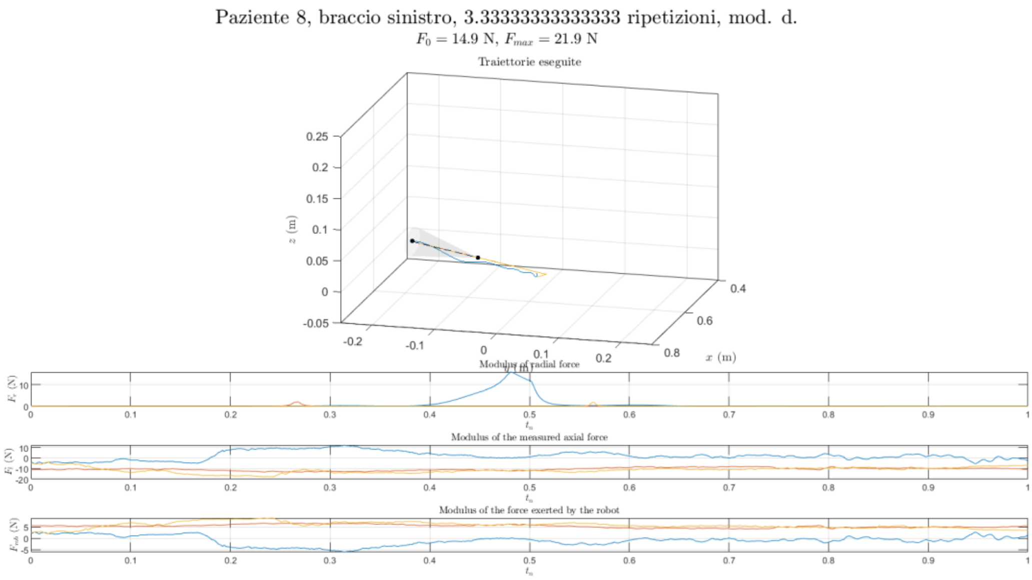


Figure 47 - Figure 48 - Force analysis related to pathological patient 8. Lef and right side in M and D-Modality.



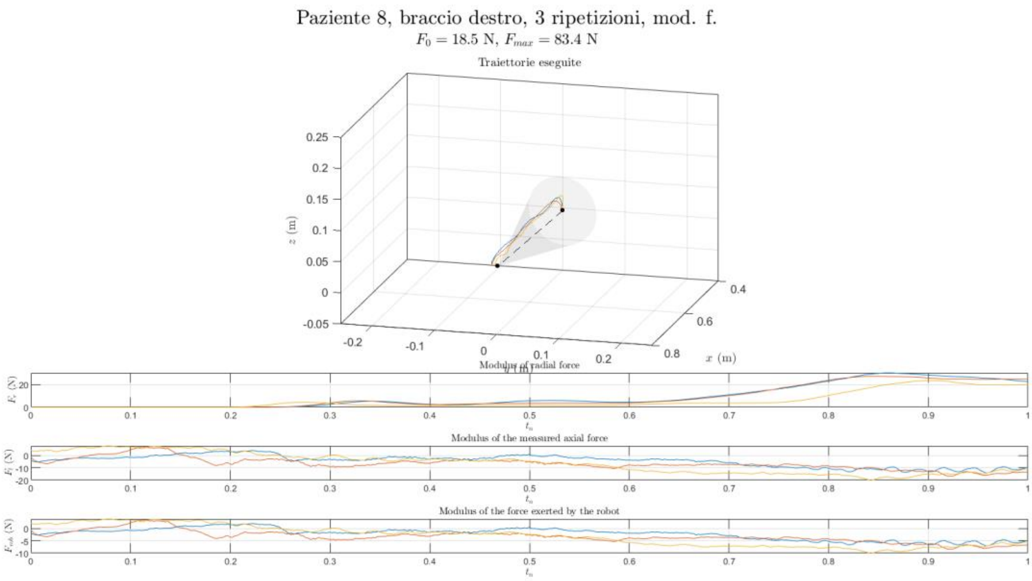


Figure 49 - Force analysis related to pathological patient 8. Lef and right side in M and D-Modality

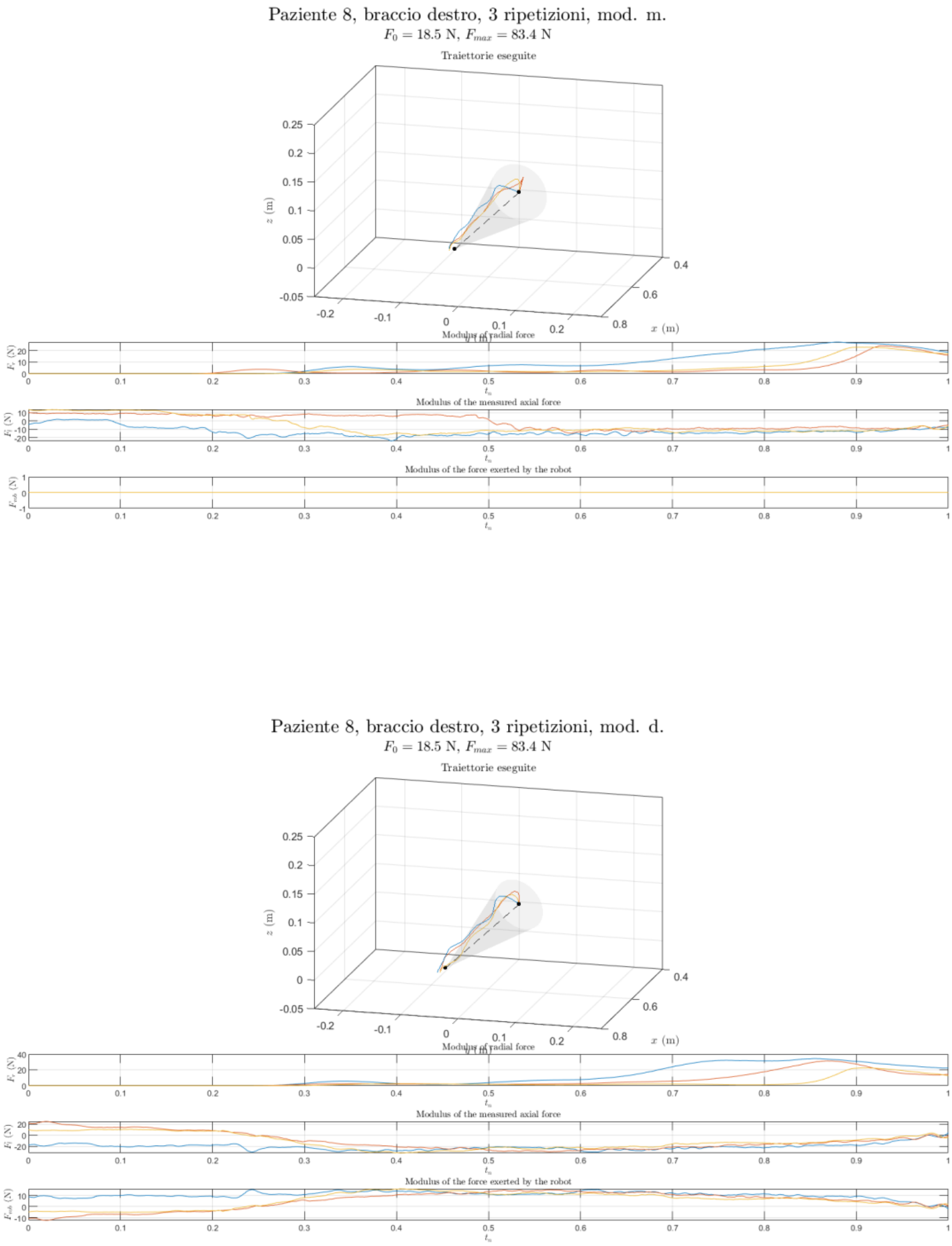
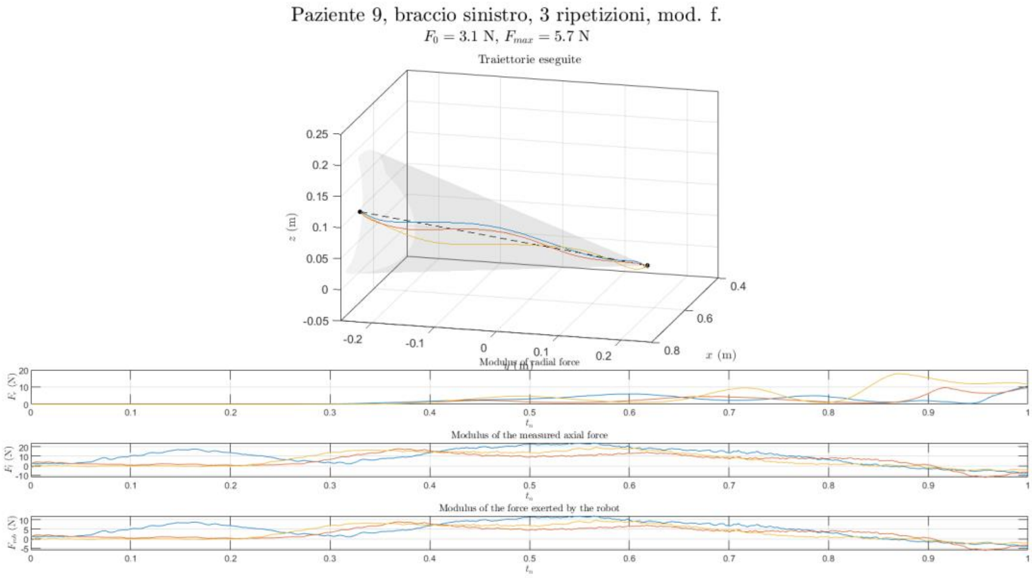


Figure 50 - Force analysis related to pathological patient 8. Lef and right side in M and D-Modality

Pathological patient 9



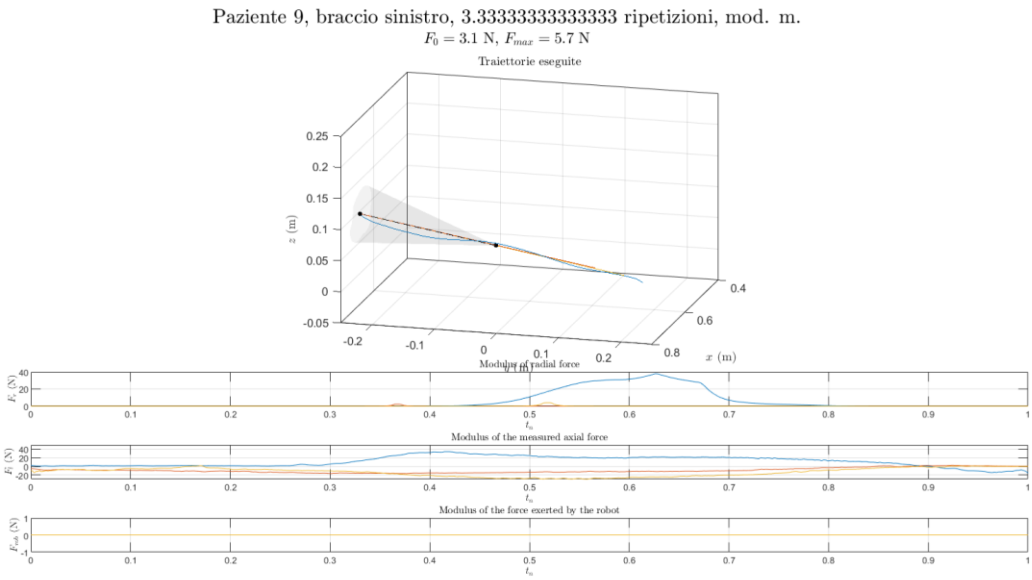


Figure 51 - Force analysis related to pathological patient 9. Lef and right side in M and D-Modality

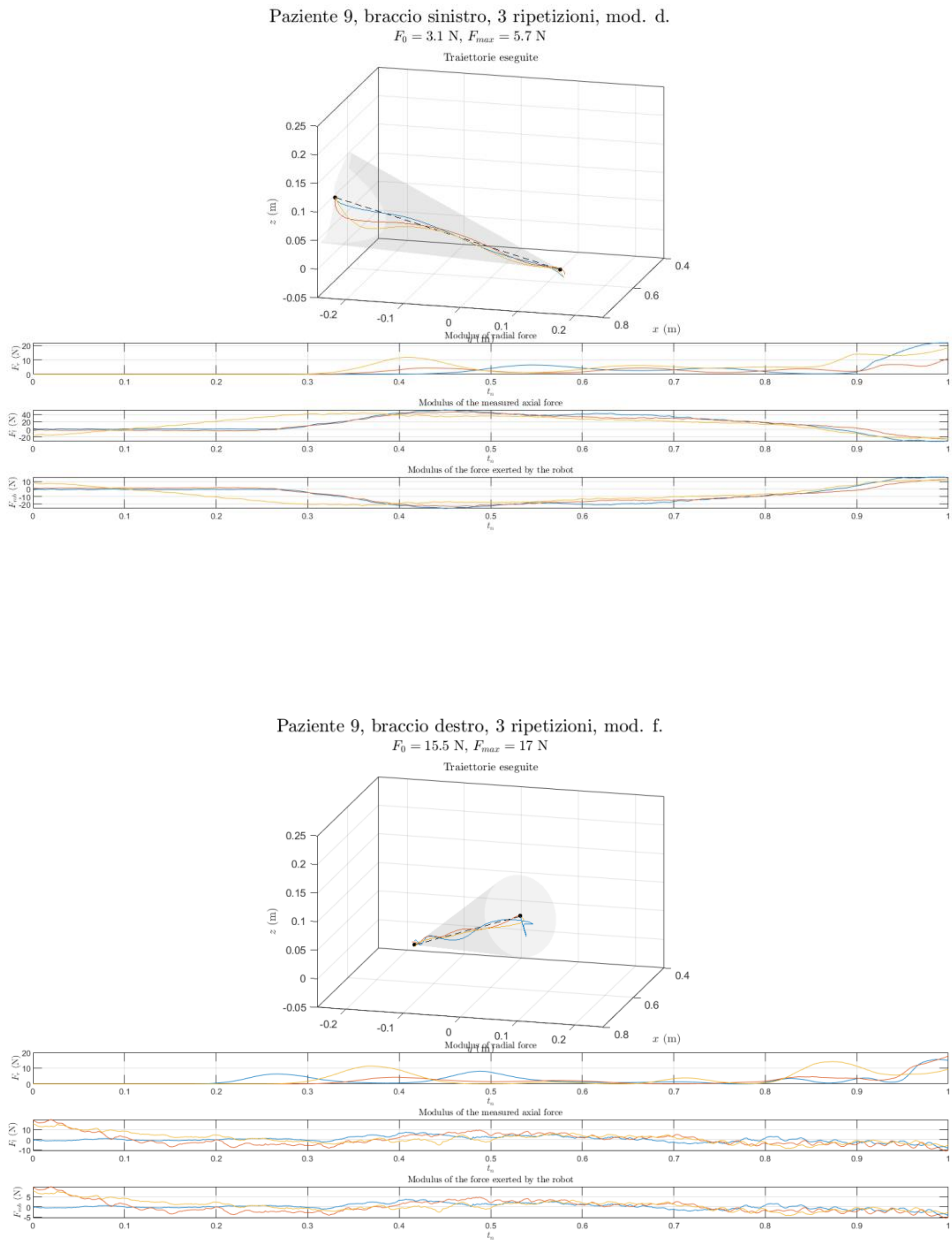
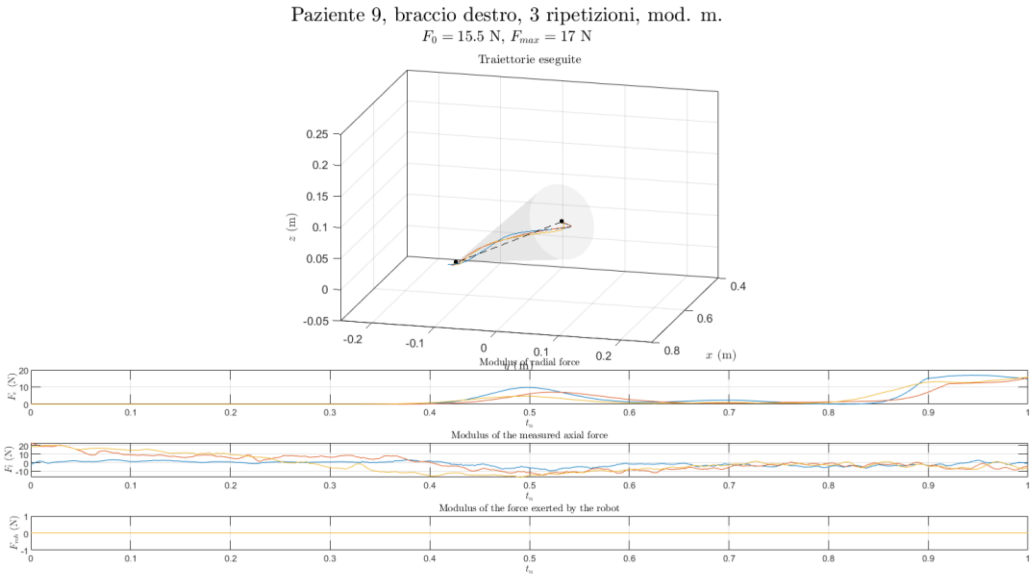


Figure 52 - Force analysis related to pathological patient 9. Lef and right side in M and D-Modality



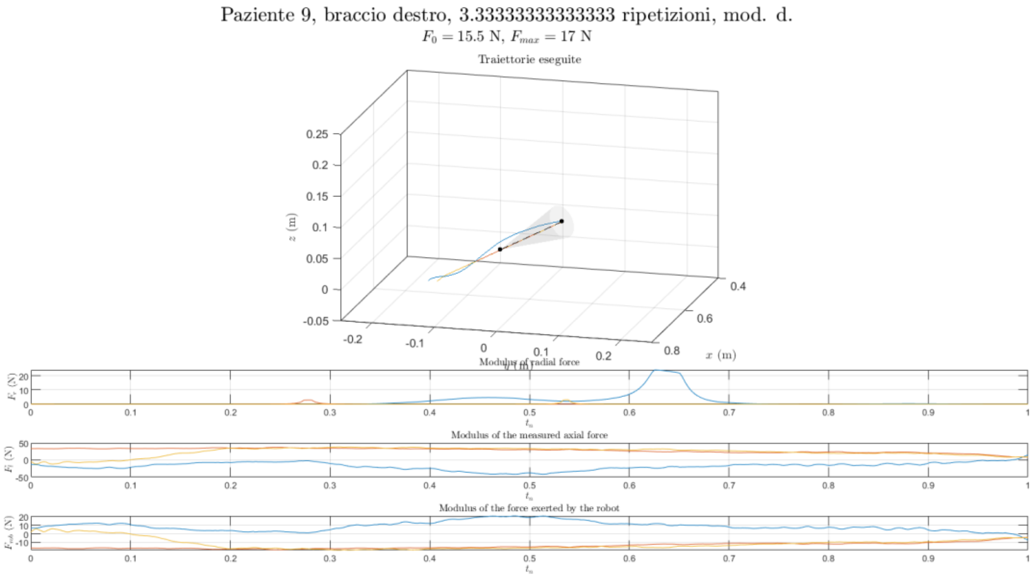


Figure 53 - Force analysis related to pathological patient 9. Lef and right side in M and D-Modality

Pathological patient 10

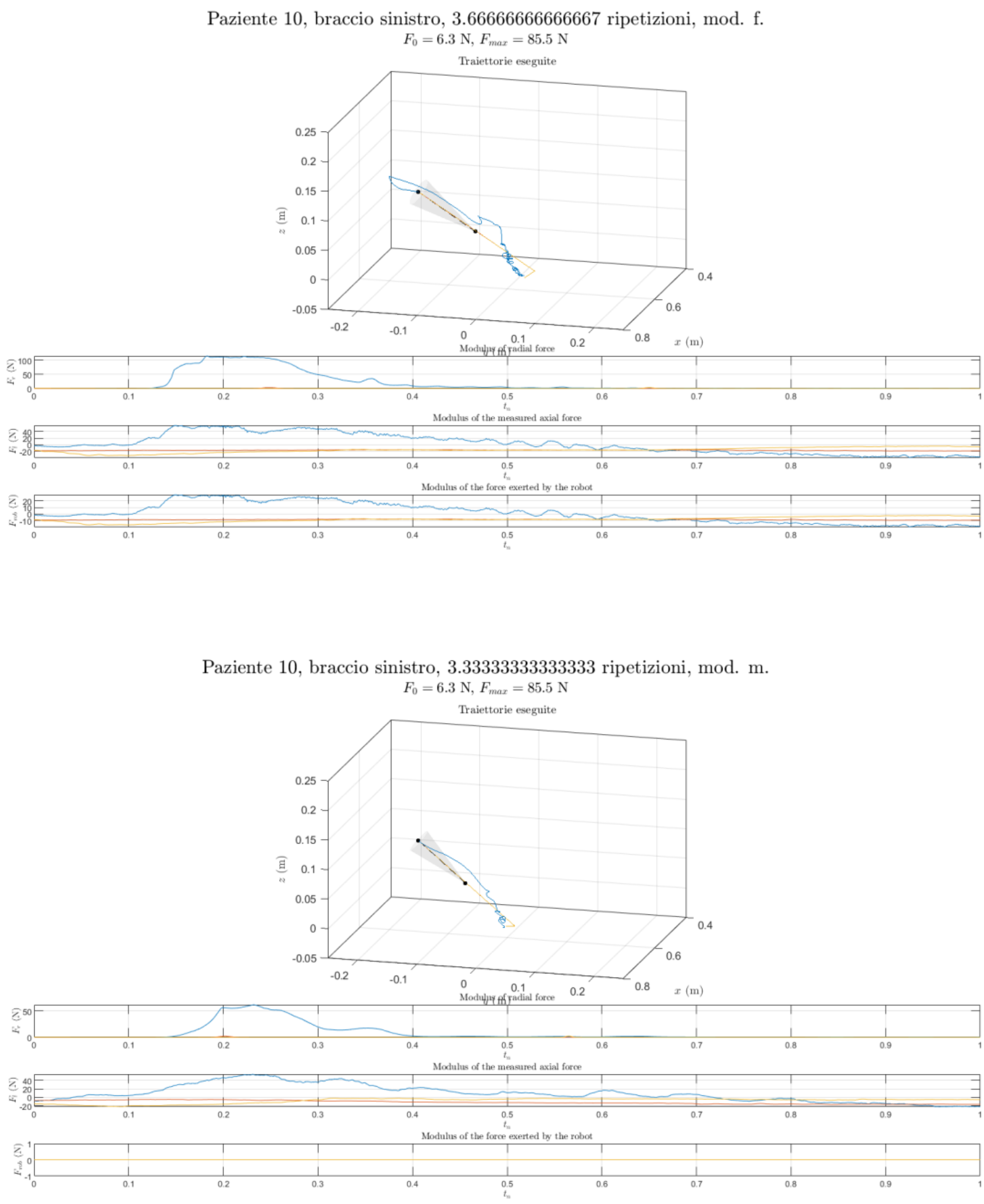
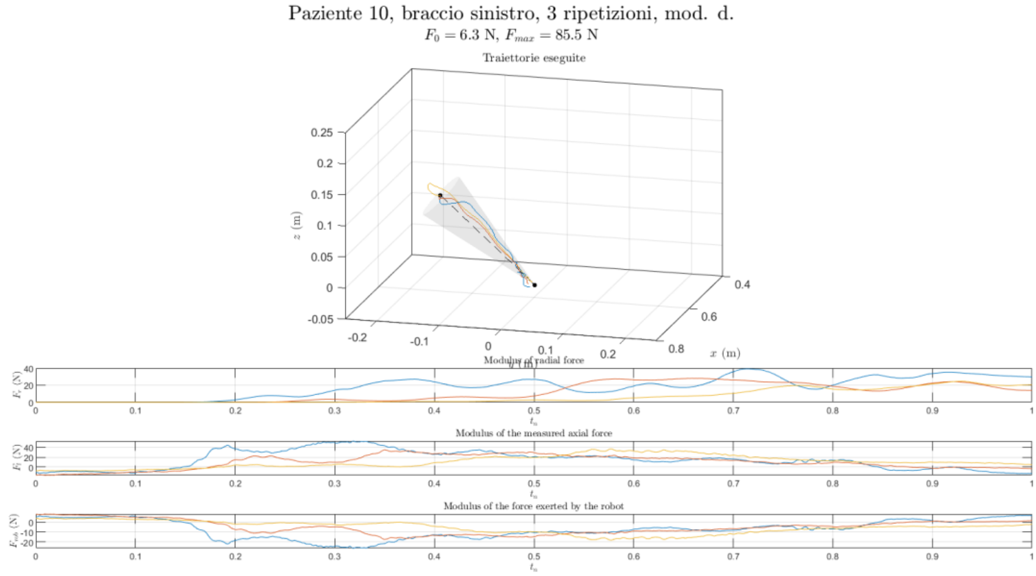


Figure 54 - Force analysis related to pathological patient 10. Lef and right side in M and D-Modality



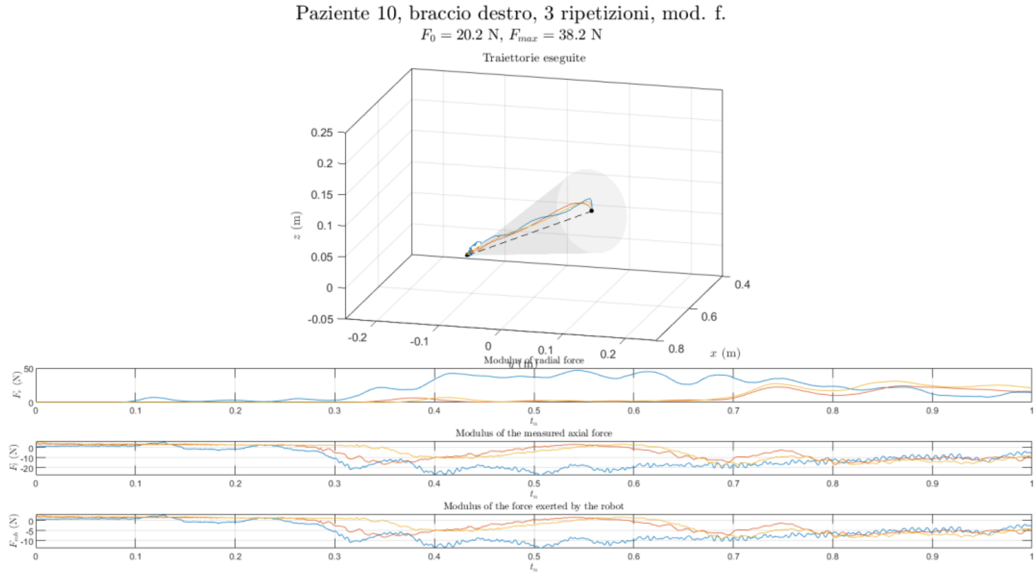


Figure 55 - Force analysis related to pathological patient 10. Lef and right side are here compared.

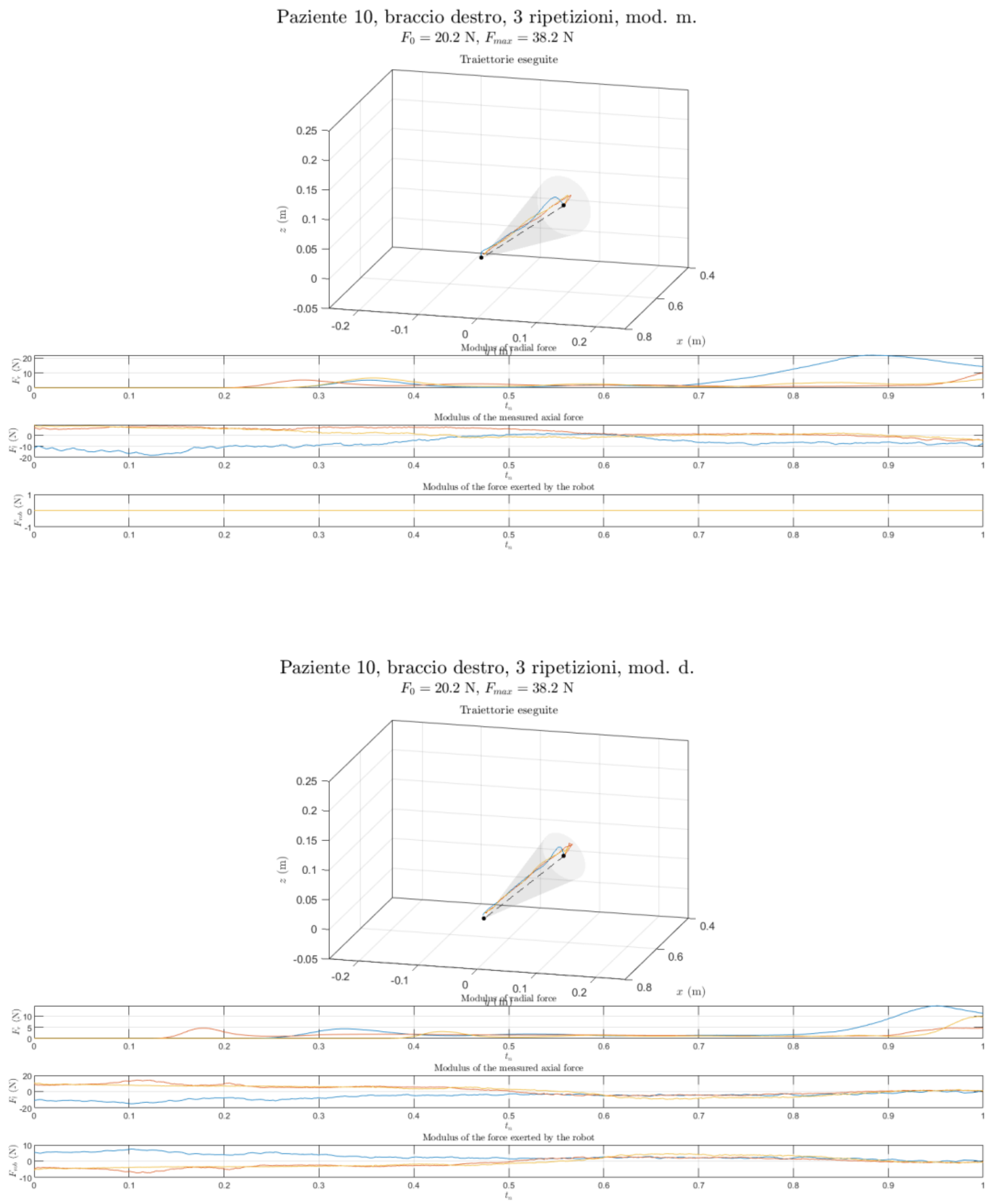
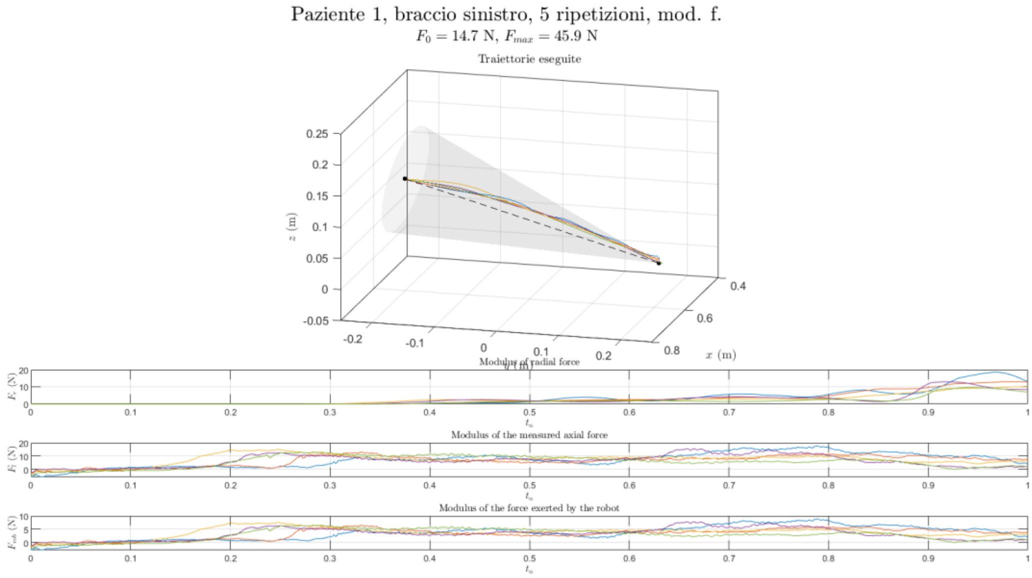
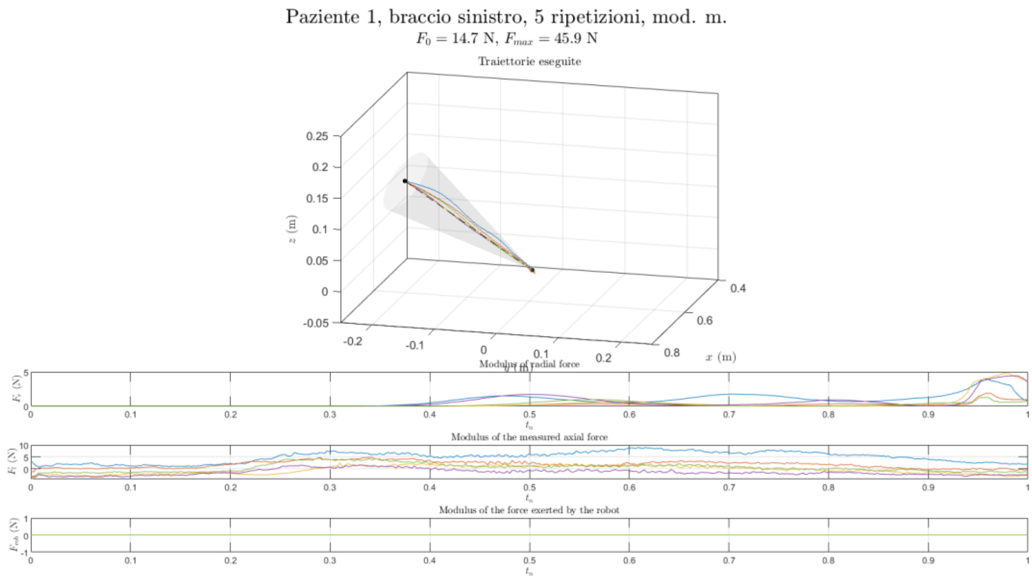
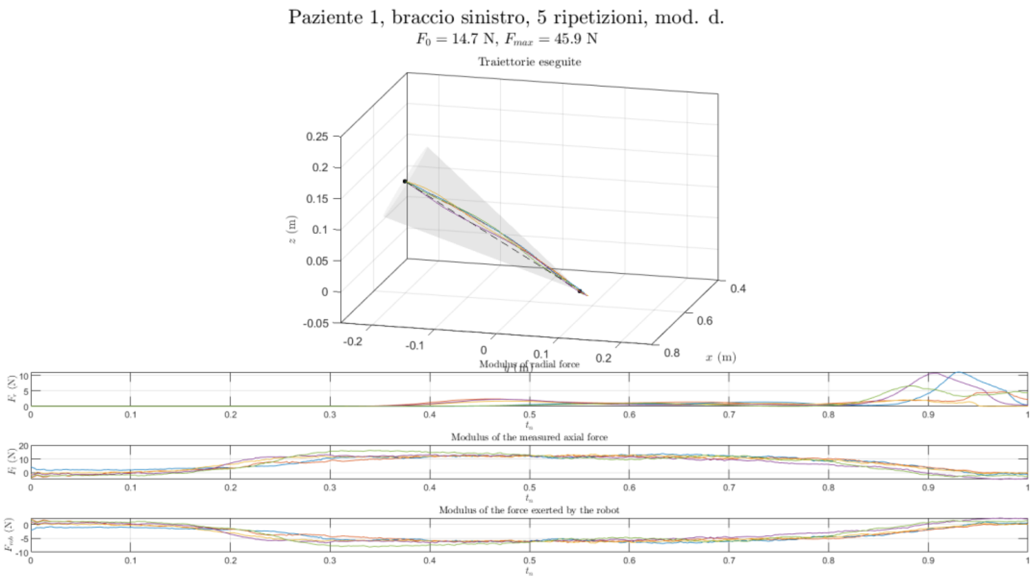


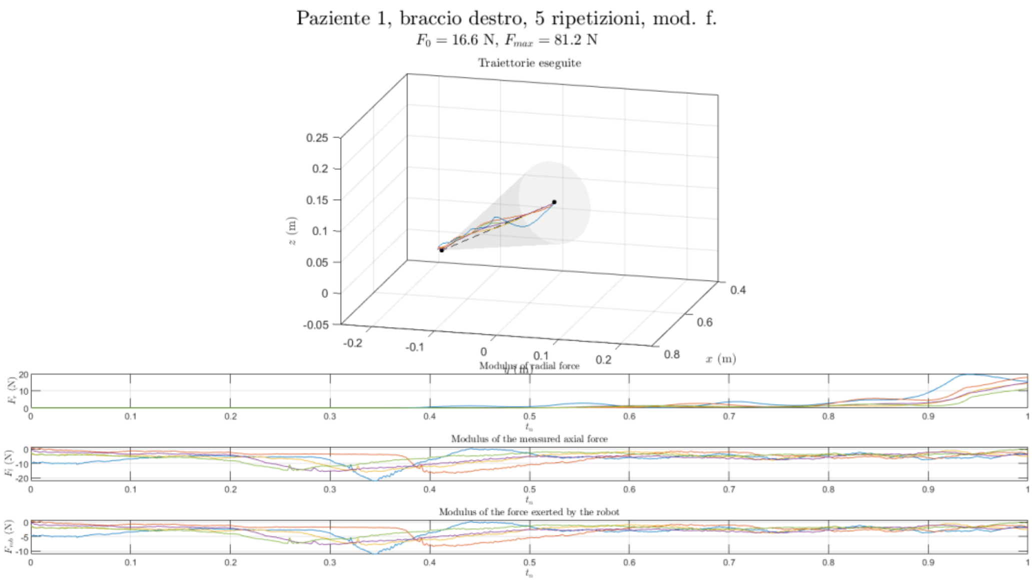
Figure 56 - Force analysis related to pathological patient 10. Lef and right side are here compared

Healthy patient 11






Paziente 1, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 14.7$ N, $F_{max} = 45.9$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 1, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 16.6$ N, $F_{max} = 81.2$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot

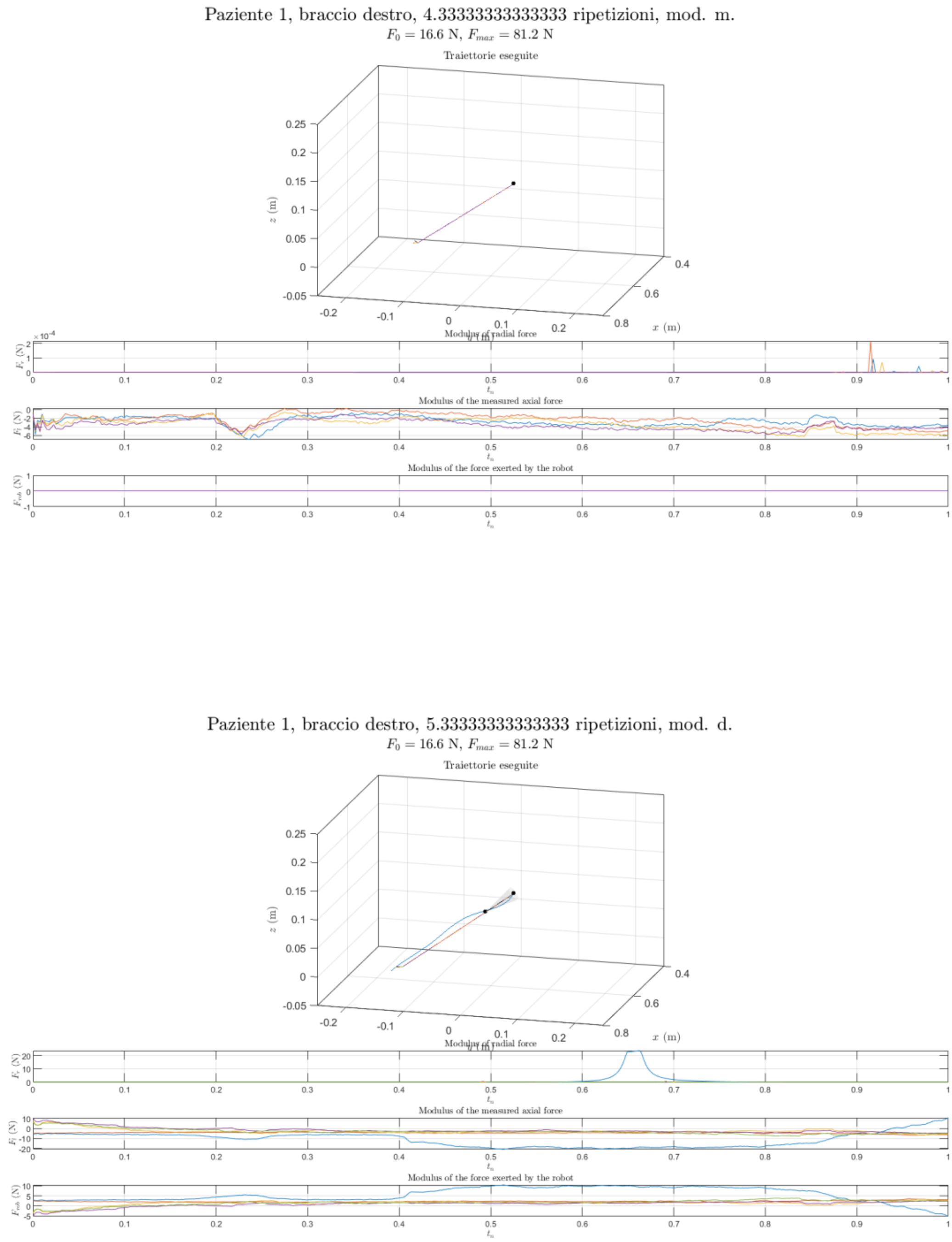
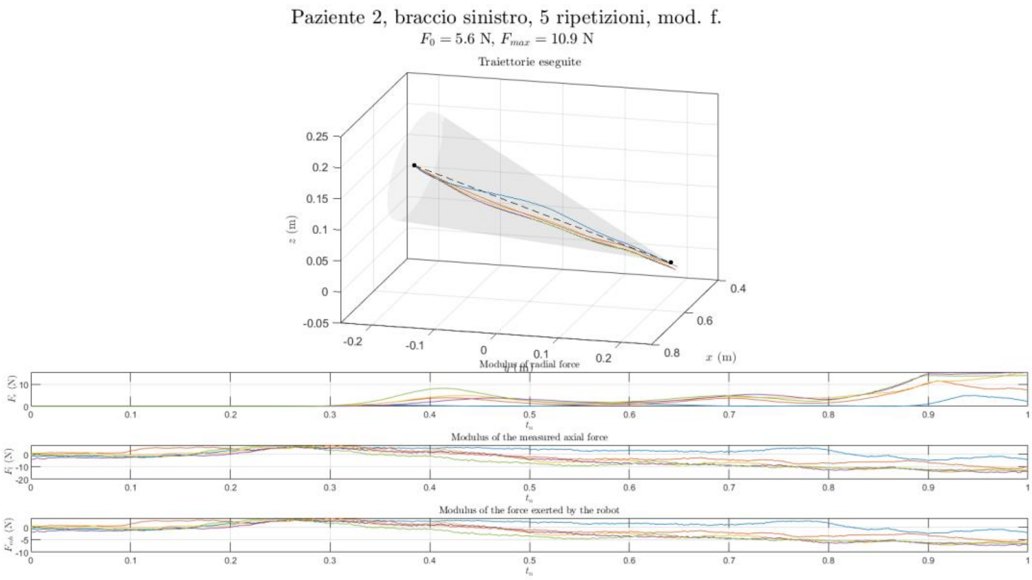
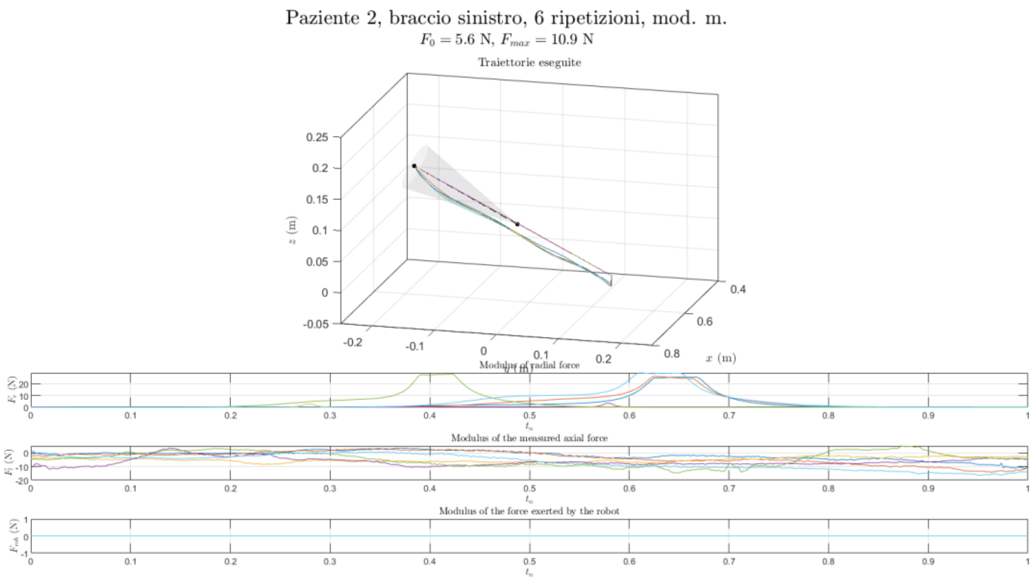
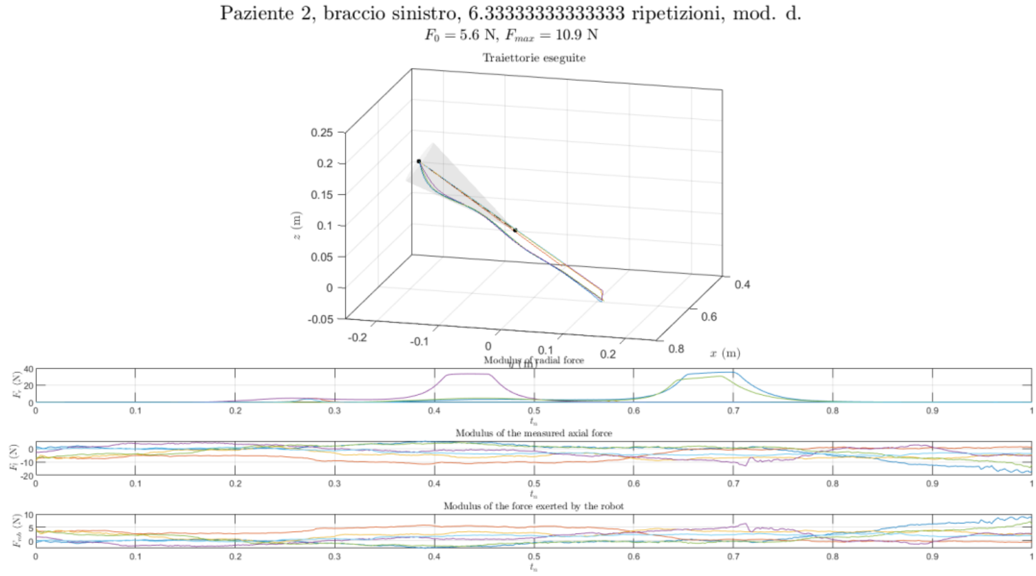


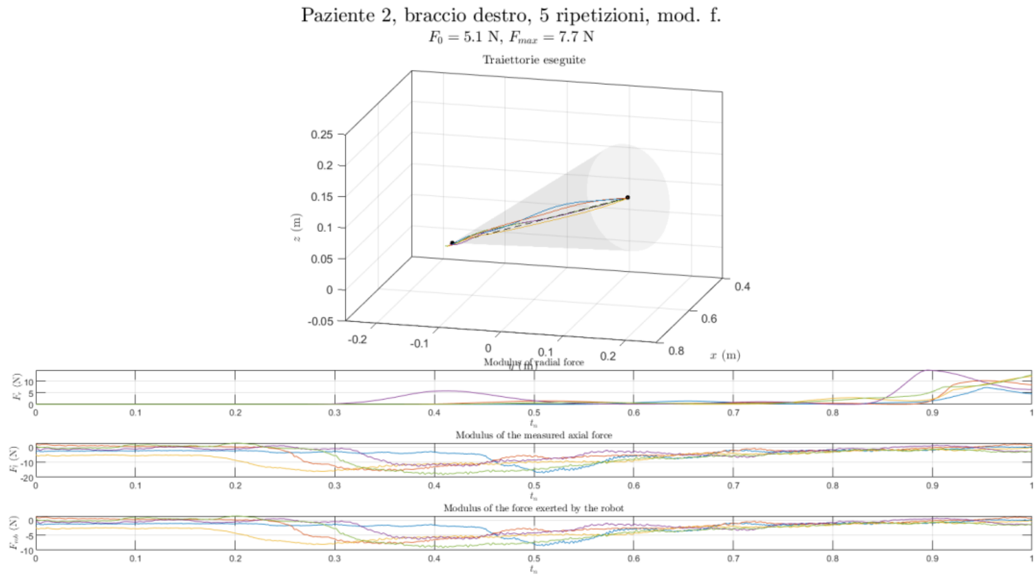
Figure 57 - Force analysis related to healthy patient 11. Lef and right side are here compared

PATIENT 12


Paziente 2, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 5.6$ N, $F_{max} = 10.9$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 2, braccio sinistro, 6 ripetizioni, mod. m.
$F_0 = 5.6$ N, $F_{max} = 10.9$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 2, braccio sinistro, 6.33333333333333 ripetizioni, mod. d.
$F_0 = 5.6$ N, $F_{max} = 10.9$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 2, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 5.1$ N, $F_{max} = 7.7$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot

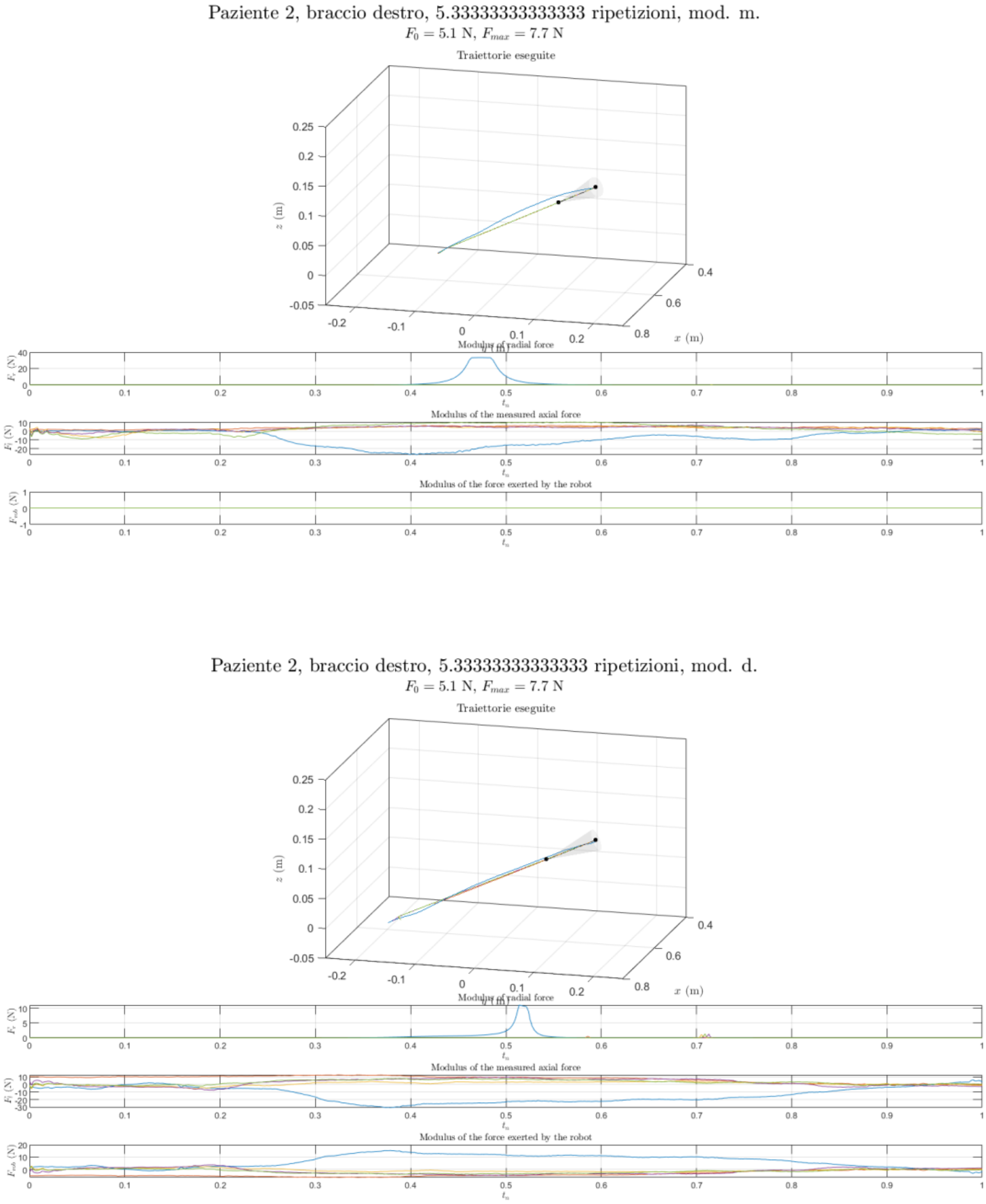
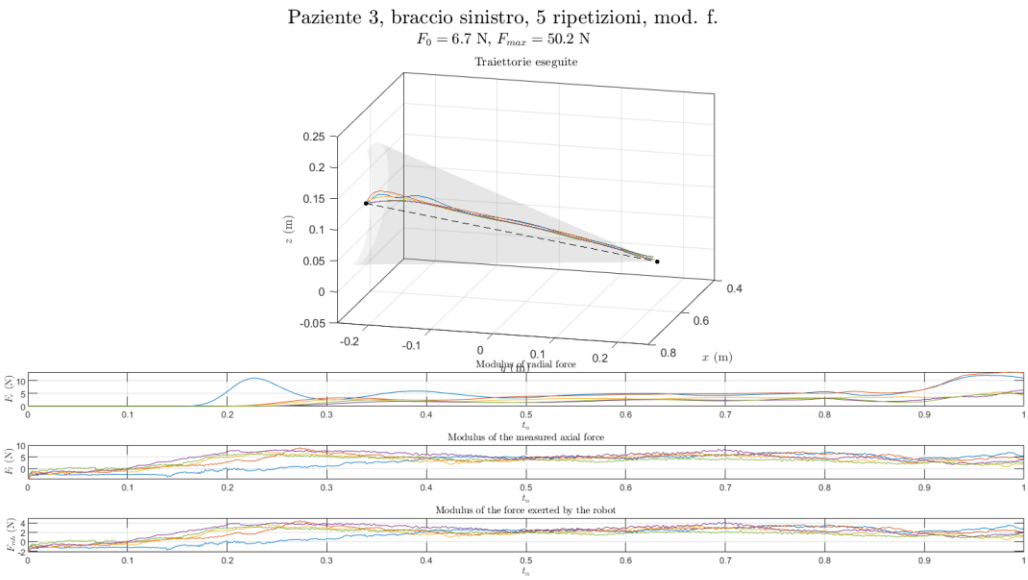
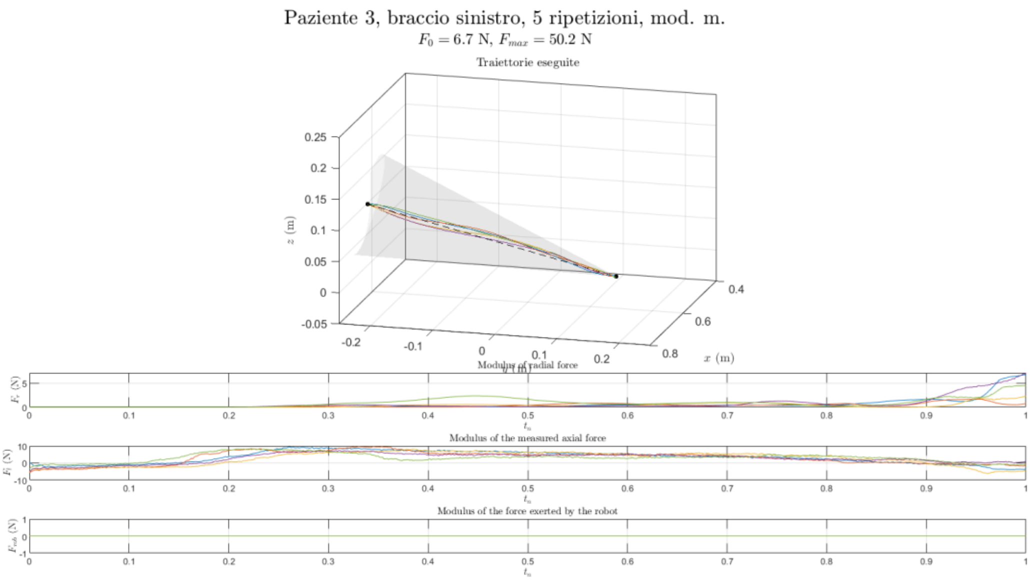
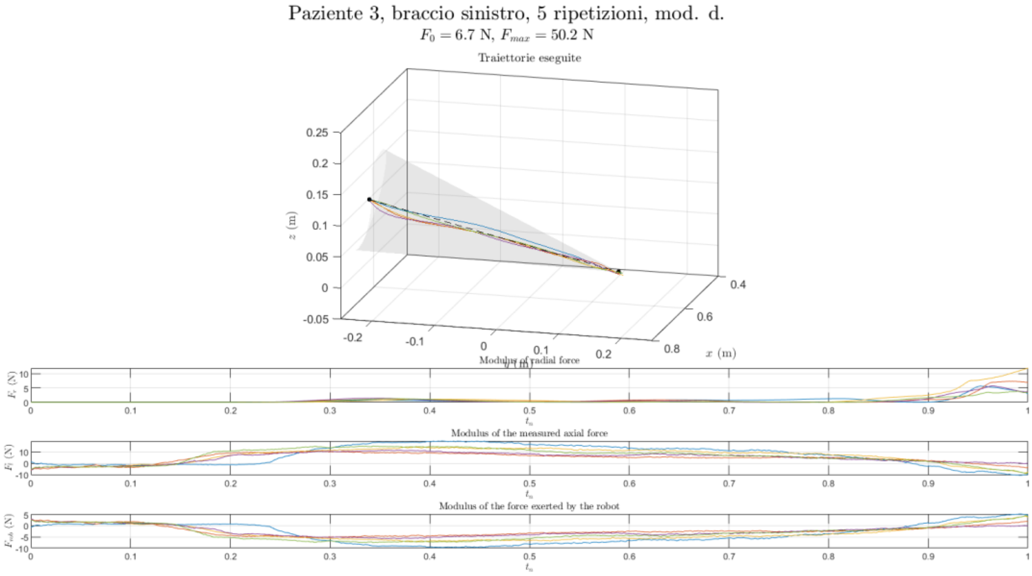


Figure 58 - Force analysis related to healthy patient 12. Lef and right side are here compared.

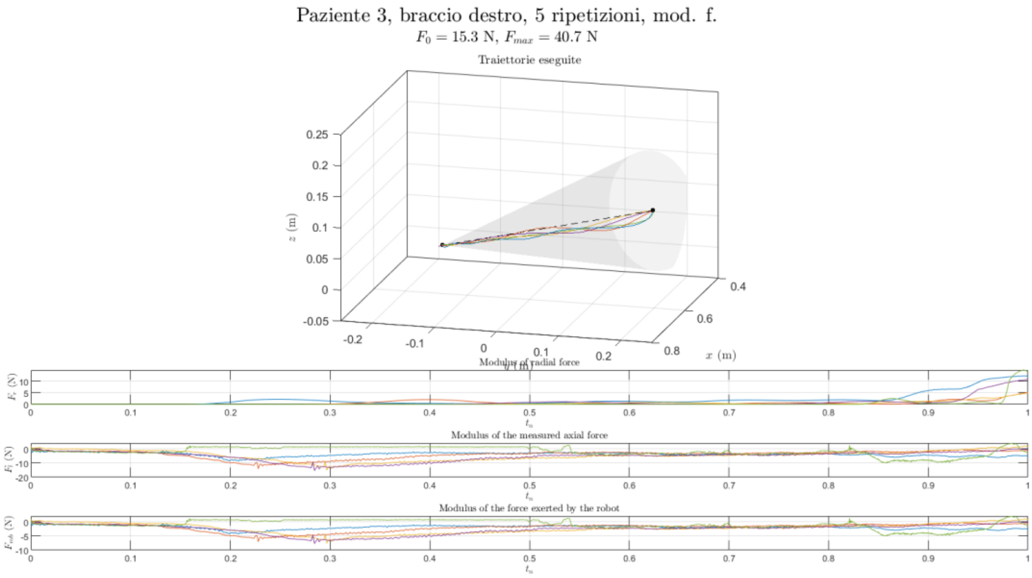
PATIENT 13


Paziente 3, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 6.7$ N, $F_{max} = 50.2$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 3, braccio sinistro, 5 ripetizioni, mod. m.
$F_0 = 6.7$ N, $F_{max} = 50.2$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot



Paziente 3, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 6.7$ N, $F_{max} = 50.2$ N
Traiettorie eseguite
z (m)
y (m)
x (m)
Modulus of radial force
F_r (N)
Modulus of the measured axial force
F_l (N)
t_n
Modulus of the force exerted by the robot
F_{rob} (N)
t_n



Paziente 3, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 15.3$ N, $F_{max} = 40.7$ N
Traiettorie eseguite
z (m)
y (m)
x (m)
Modulus of radial force
F_r (N)
Modulus of the measured axial force
F_l (N)
t_n
Modulus of the force exerted by the robot
F_{rob} (N)
t_n

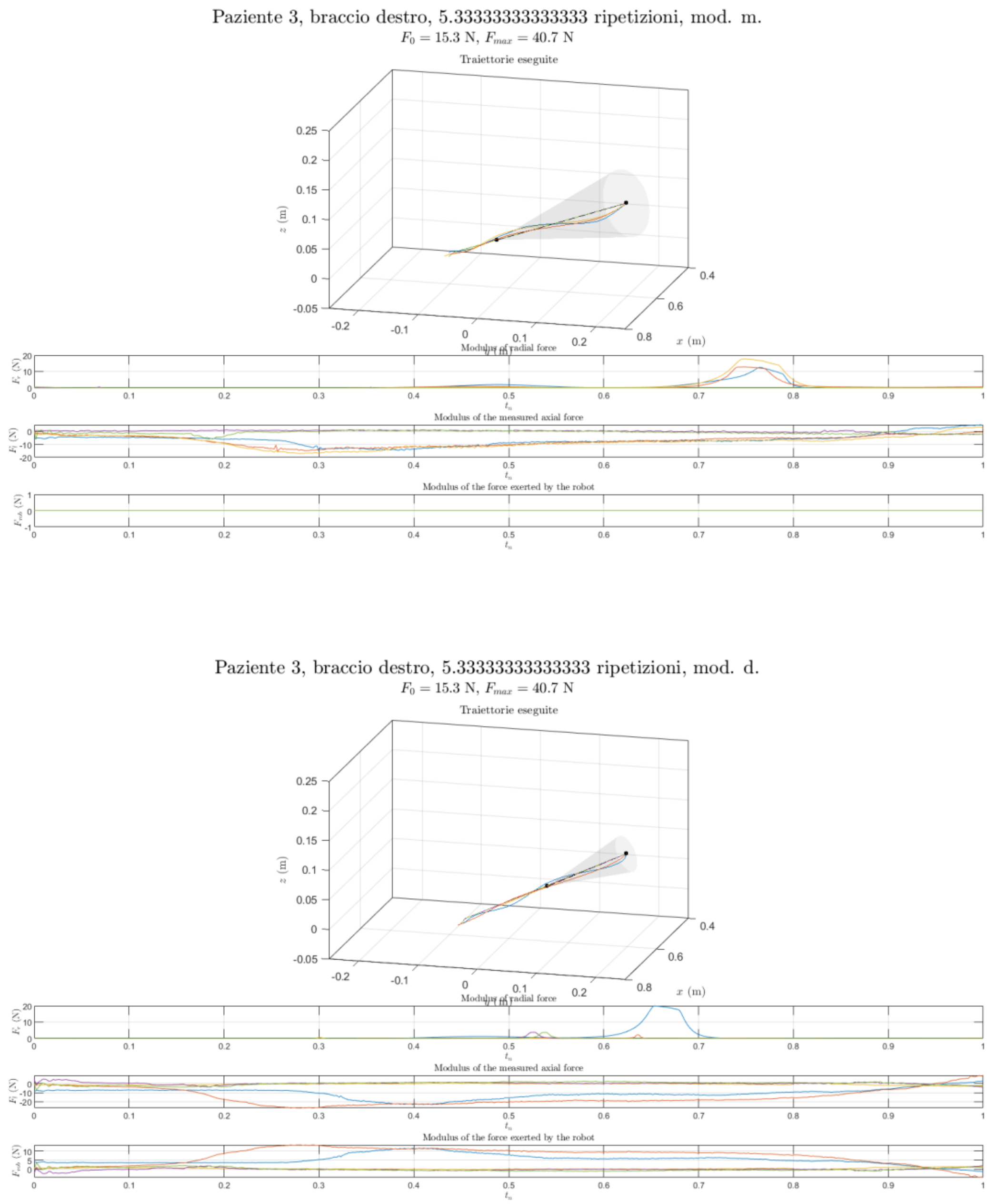
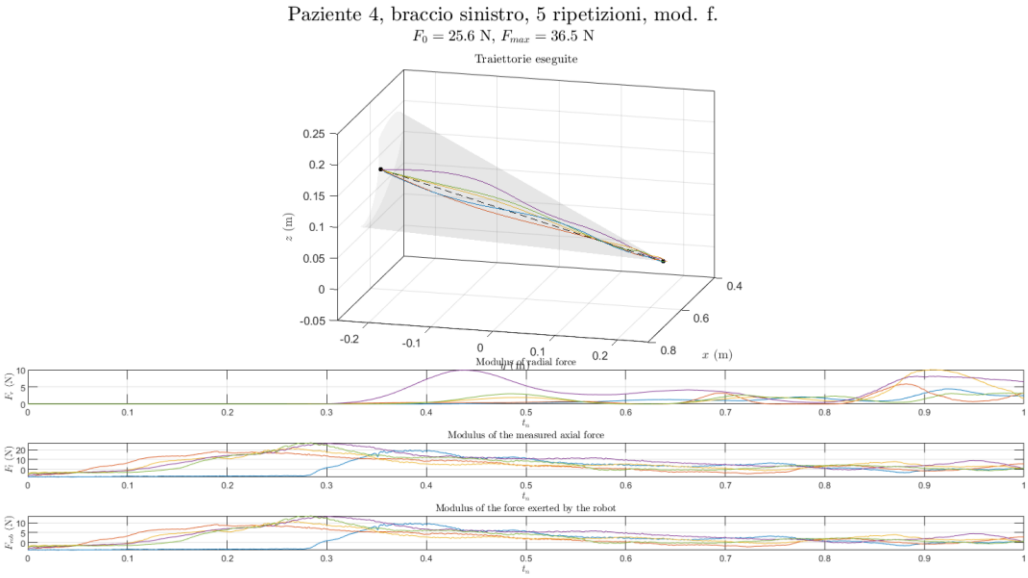
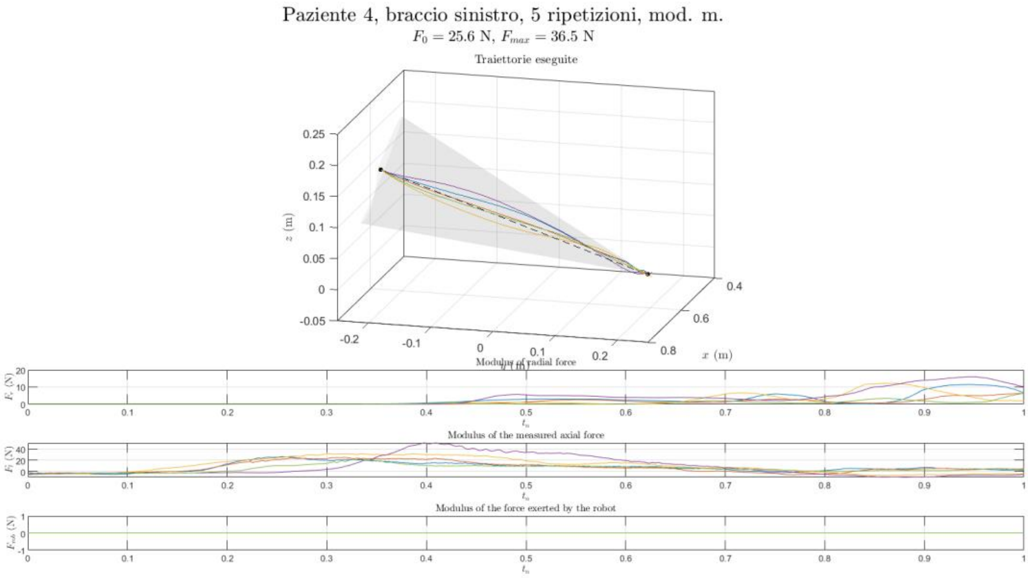


Figure 59 - Force analysis related to healthy patient 12. Lef and right side are here compared

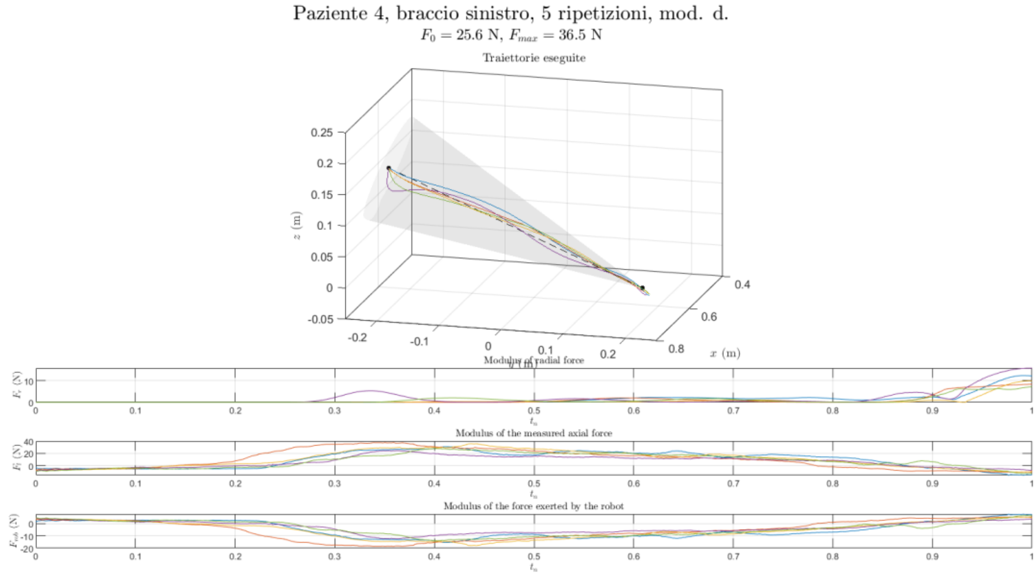
PATIENT 14

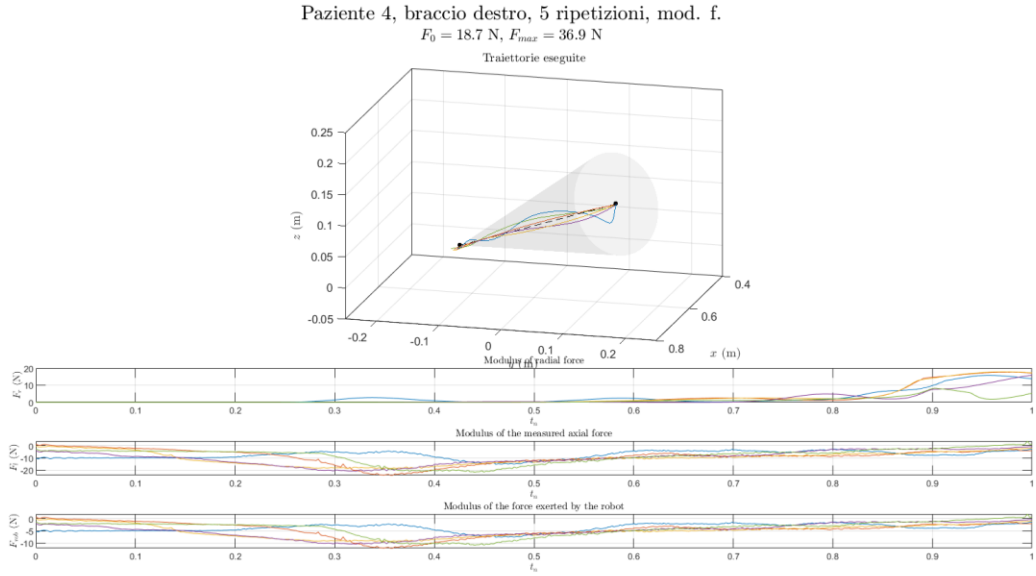


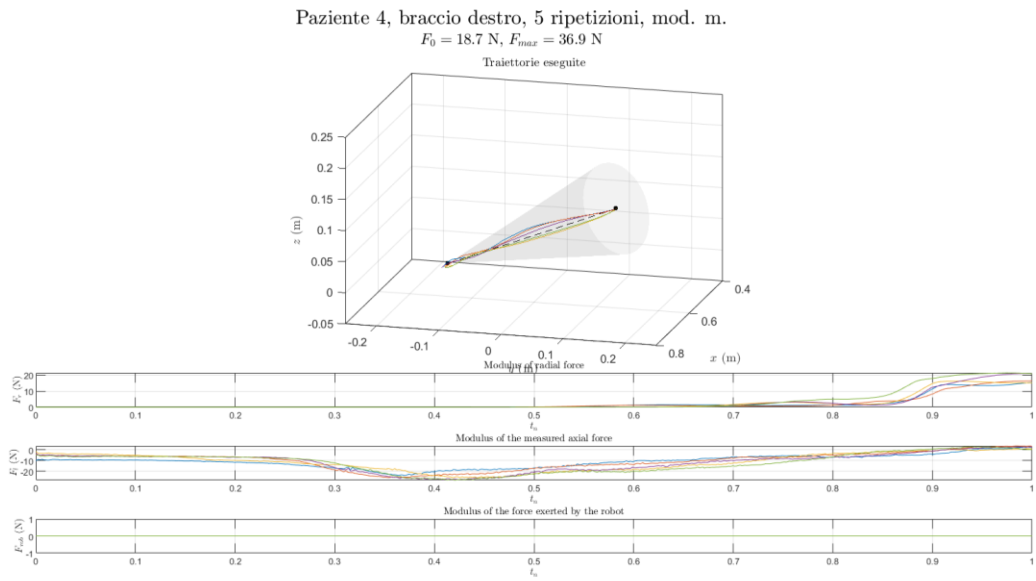
Paziente 4, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 25.6$ N, $F_{max} = 36.5$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
F_r (N)
Modulus of the measured axial force
F_t (N)
Modulus of the force exerted by the robot
F_{rob} (N)
t_n



Paziente 4, braccio sinistro, 5 ripetizioni, mod. m.
$F_0 = 25.6$ N, $F_{max} = 36.5$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
F_r (N)
Modulus of the measured axial force
F_t (N)
Modulus of the force exerted by the robot
F_{rob} (N)
t_n


Paziente 4, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 25.6$ N, $F_{max} = 36.5$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n


Paziente 4, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 18.7$ N, $F_{max} = 36.9$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n



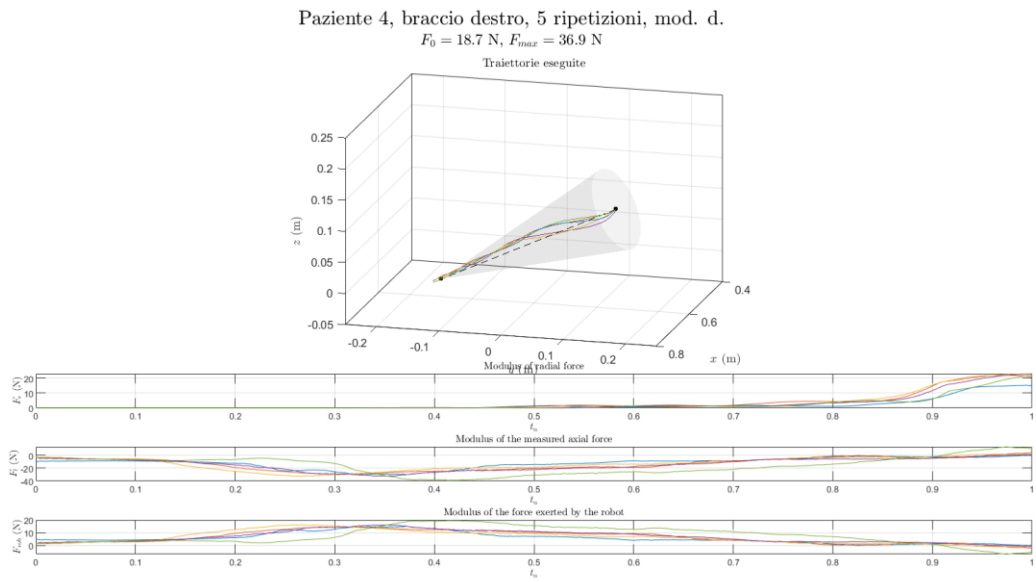
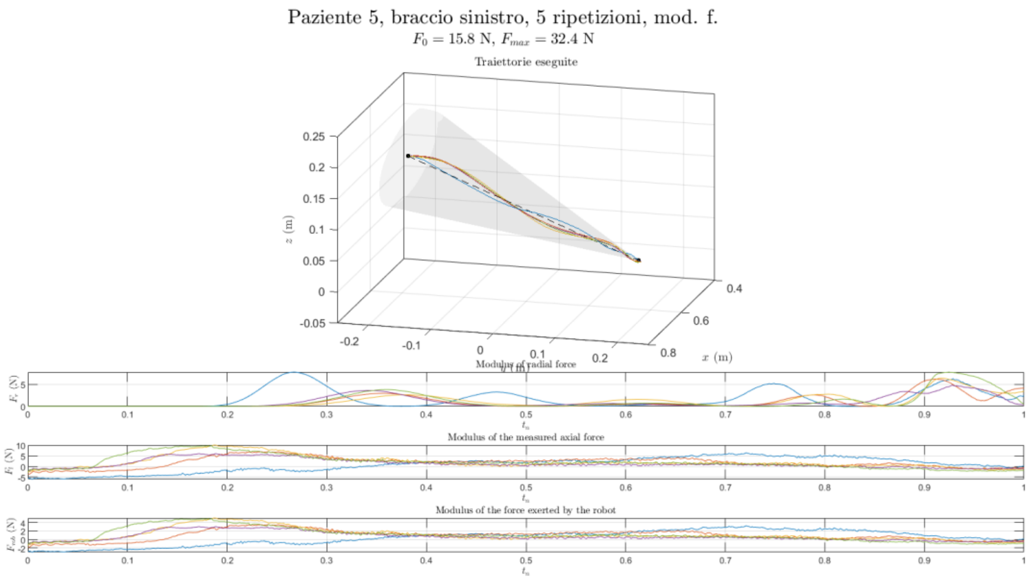
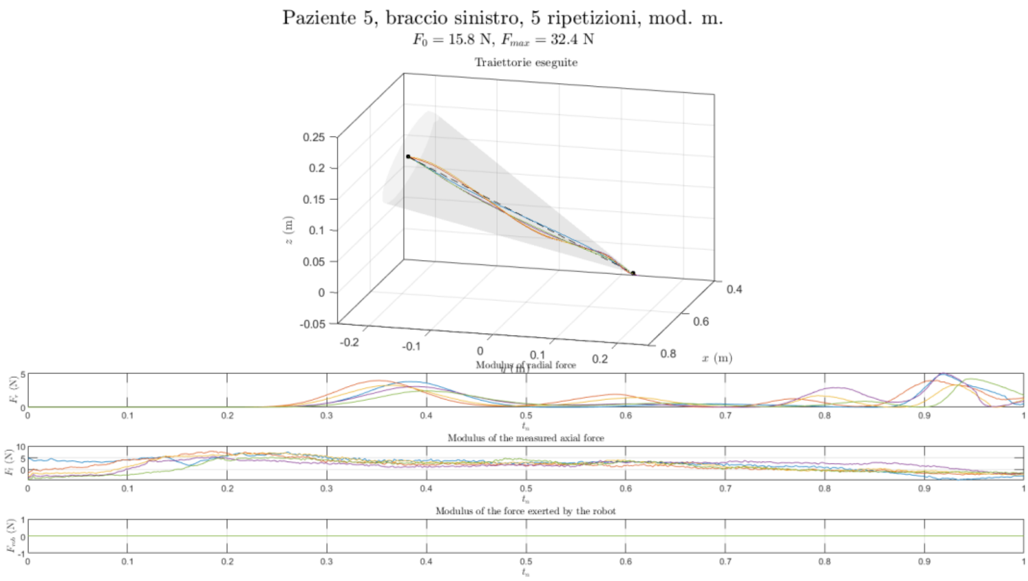
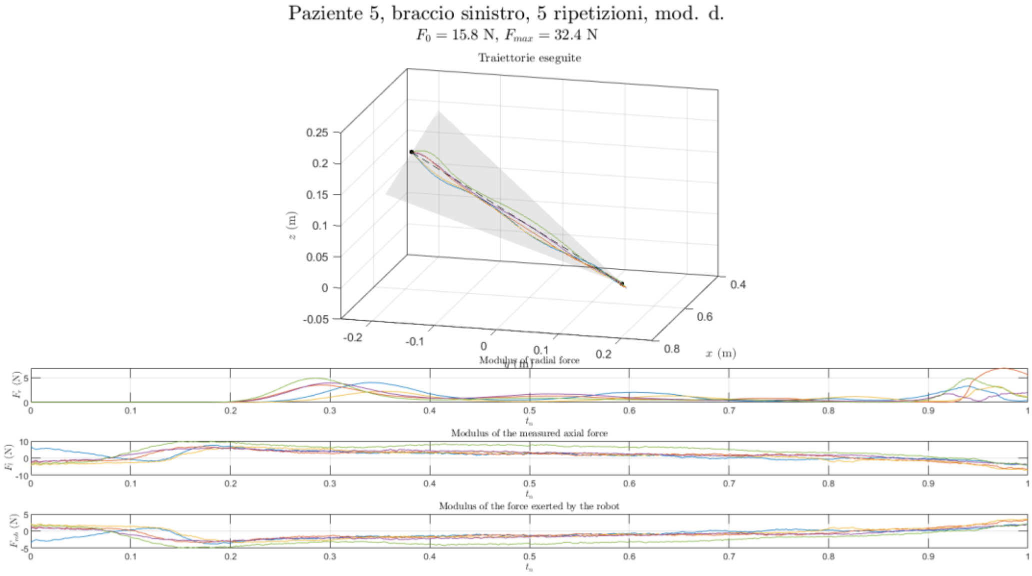


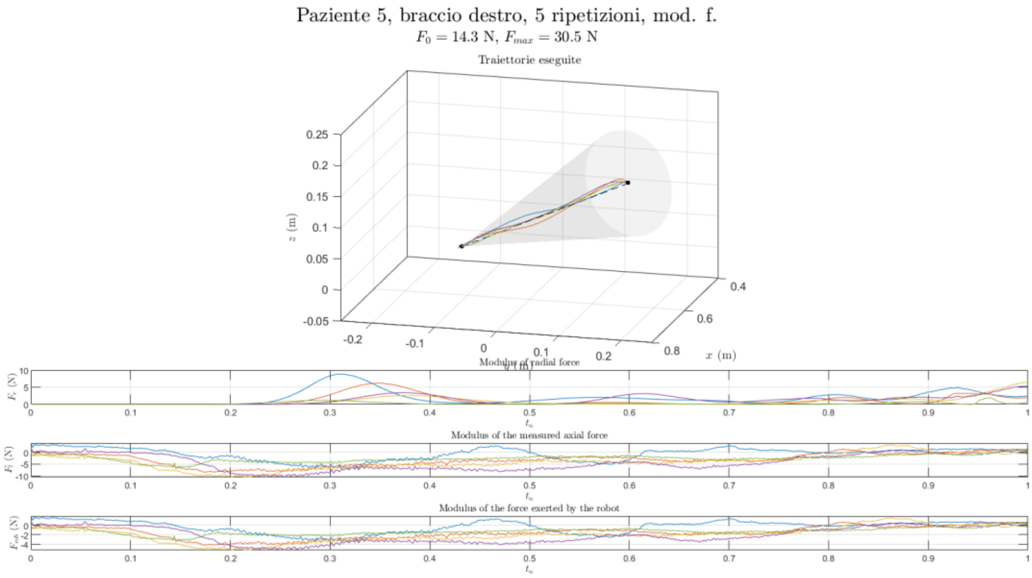
Figure 60 - Force analysis related to healthy patient 14. Lef and right side are here compared

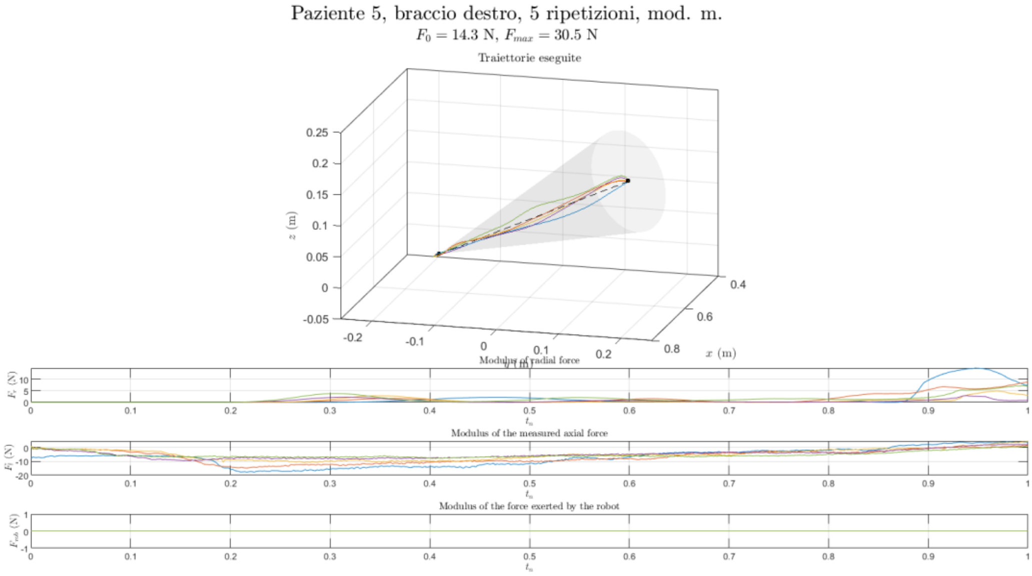
PATIENT 15


Paziente 5, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 15.8$ N, $F_{max} = 32.4$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
F_r (N)
t_n
Modulus of the measured axial force
F_l (N)
t_n
Modulus of the force exerted by the robot
F_{rob} (N)
t_n


Paziente 5, braccio sinistro, 5 ripetizioni, mod. m.
$F_0 = 15.8$ N, $F_{max} = 32.4$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
F_r (N)
t_n
Modulus of the measured axial force
F_l (N)
t_n
Modulus of the force exerted by the robot
F_{rob} (N)
t_n


Paziente 5, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 15.8$ N, $F_{max} = 32.4$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 5, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 14.3$ N, $F_{max} = 30.5$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot



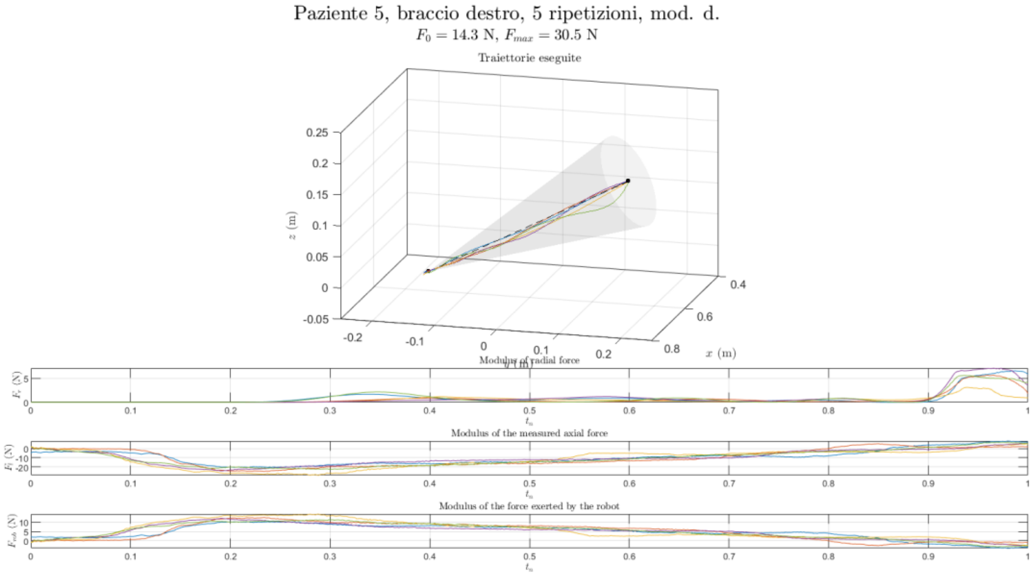
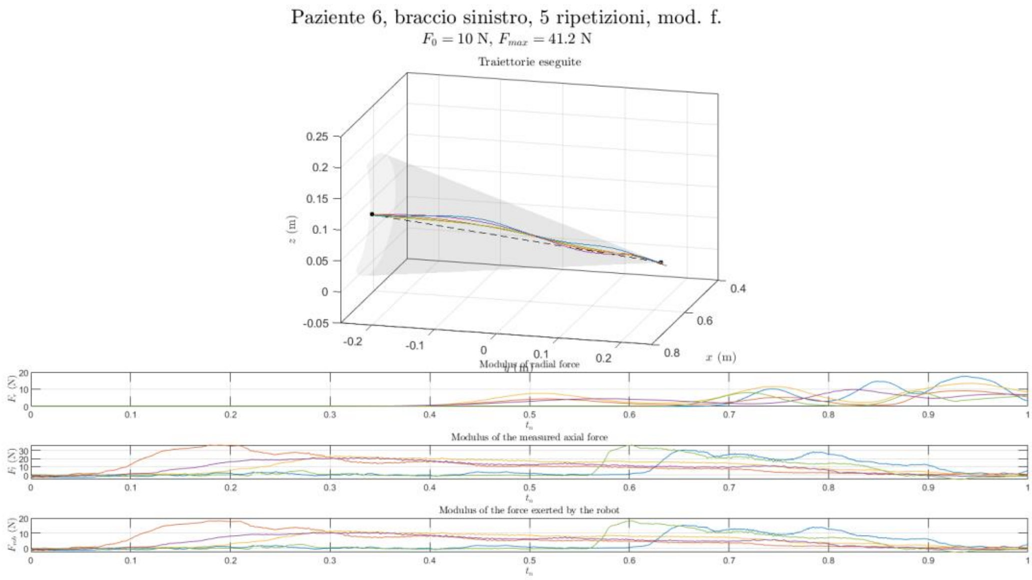
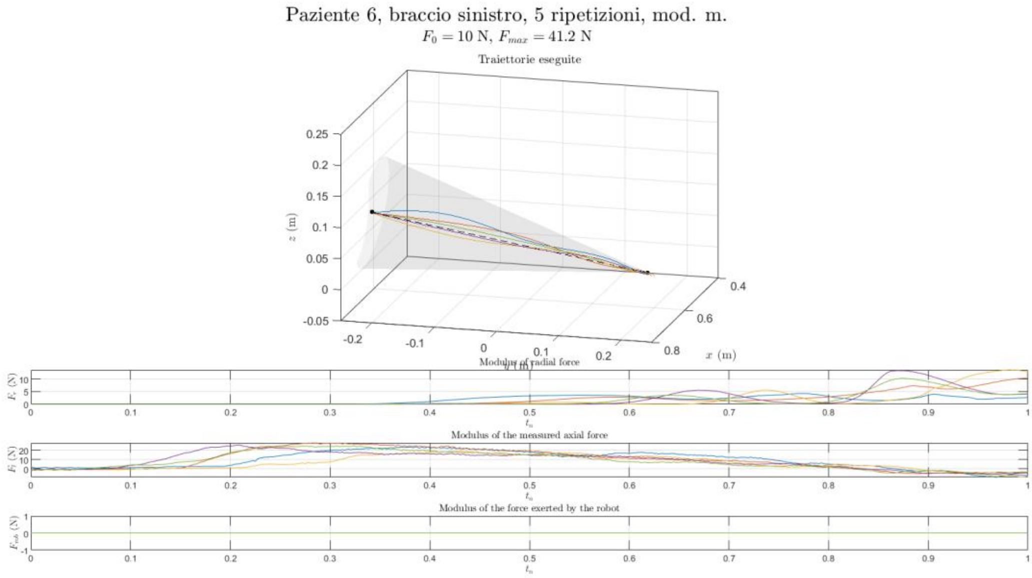
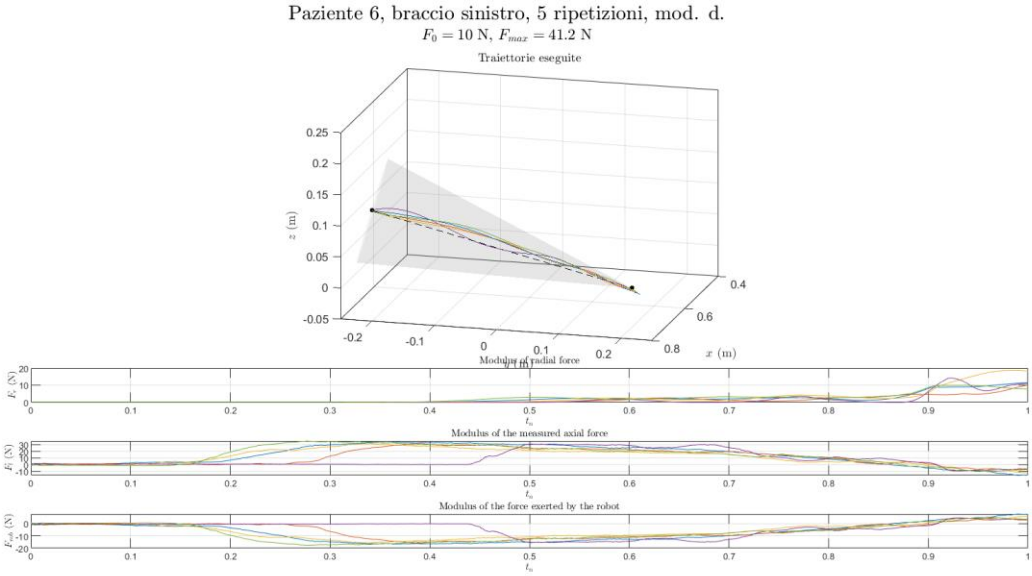


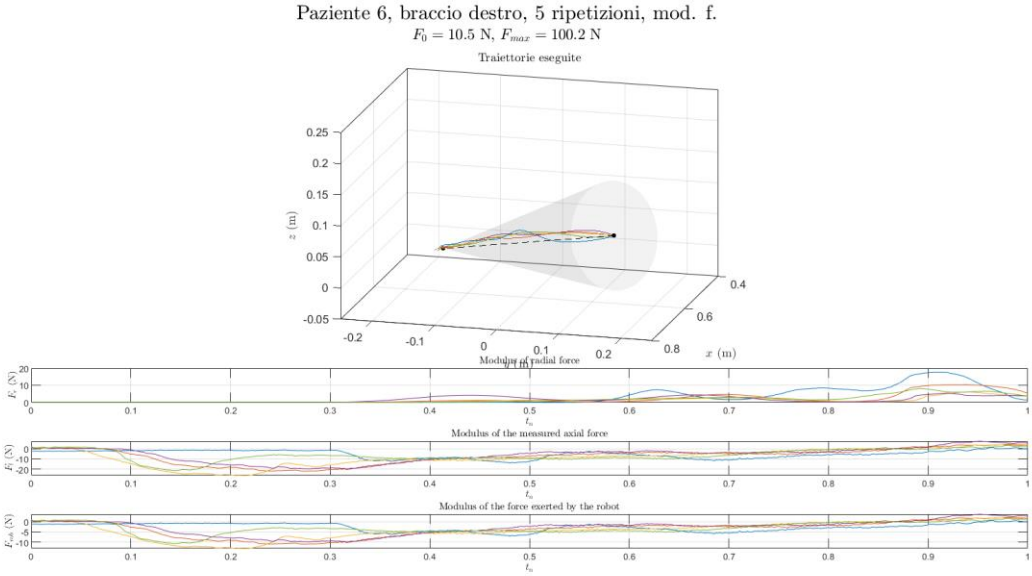
Figure 61 - Force analysis related to healthy patient 15. Lef and right side are here compared

PATIENT 16


Paziente 6, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 10$ N, $F_{max} = 41.2$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n


Paziente 6, braccio sinistro, 5 ripetizioni, mod. m.
$F_0 = 10$ N, $F_{max} = 41.2$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n


Paziente 6, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 10$ N, $F_{max} = 41.2$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 6, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 10.5$ N, $F_{max} = 100.2$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot

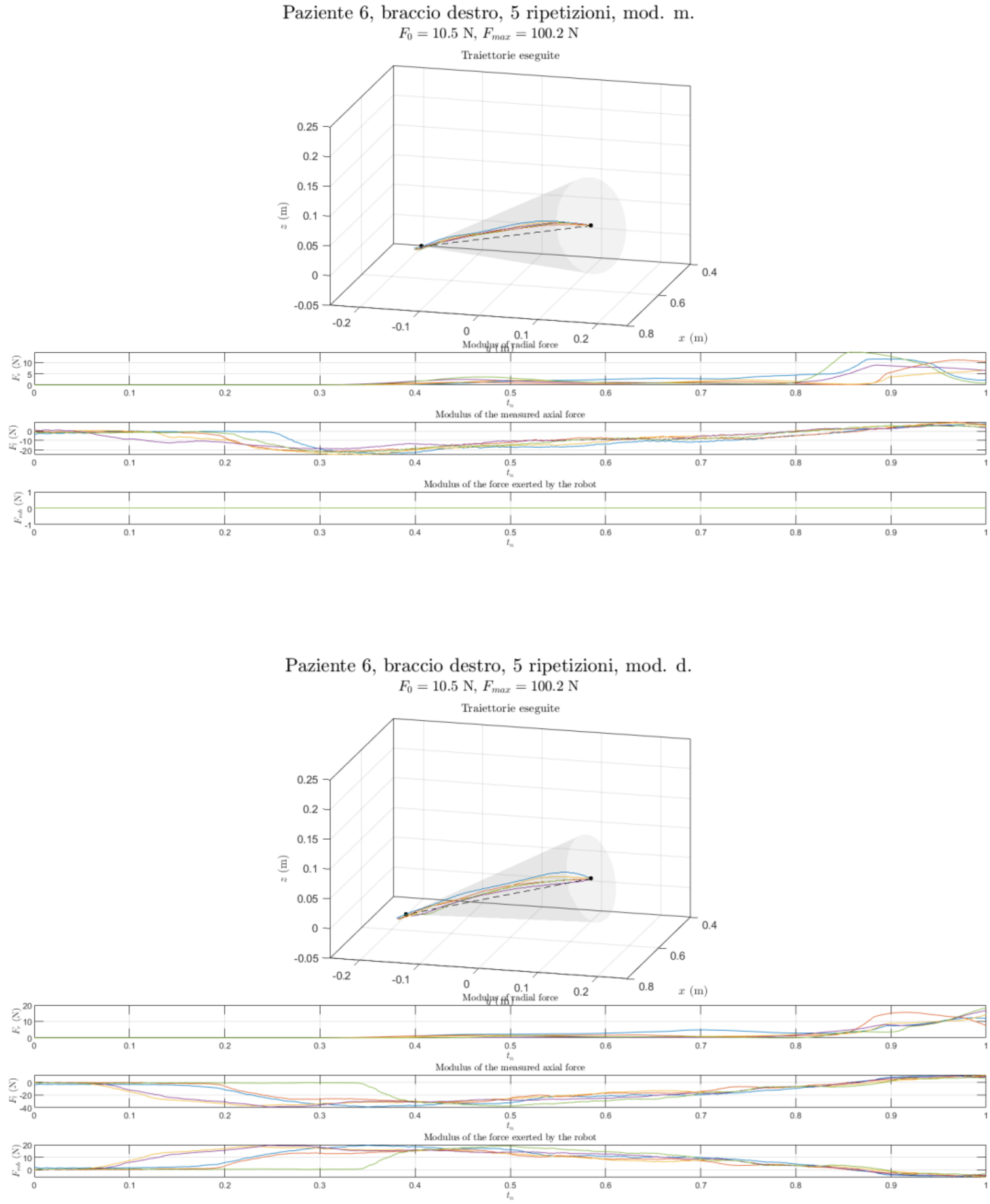
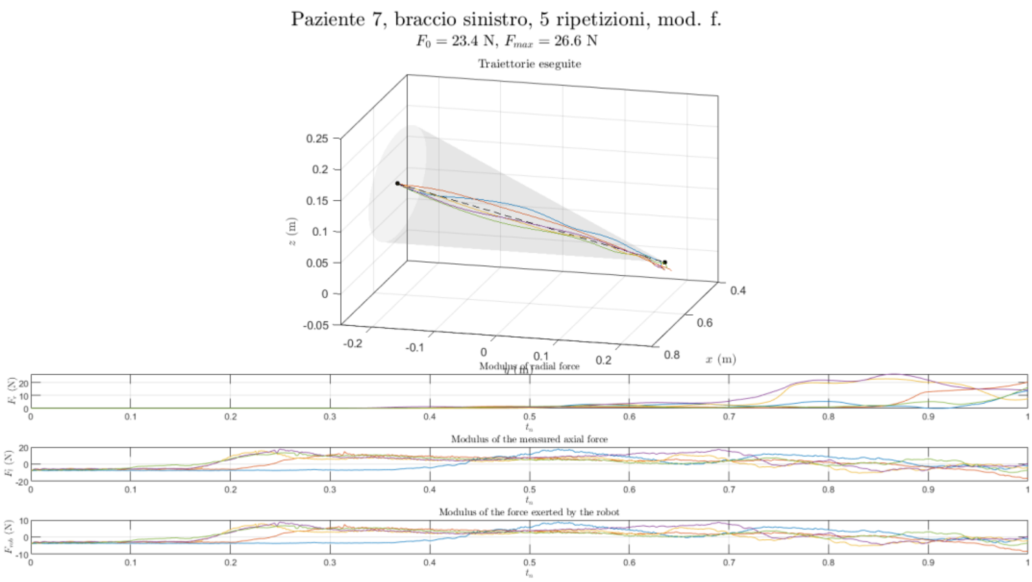
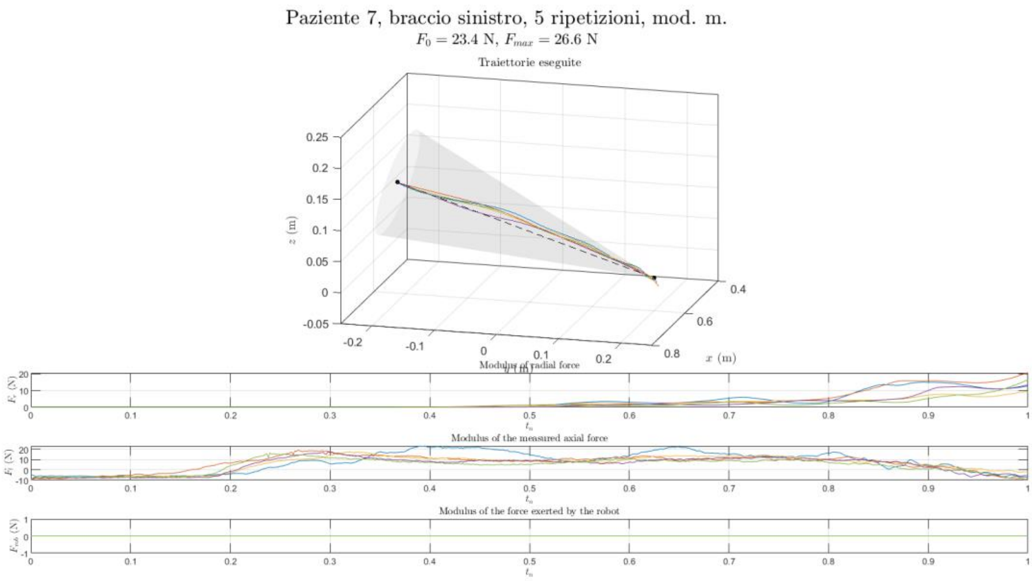
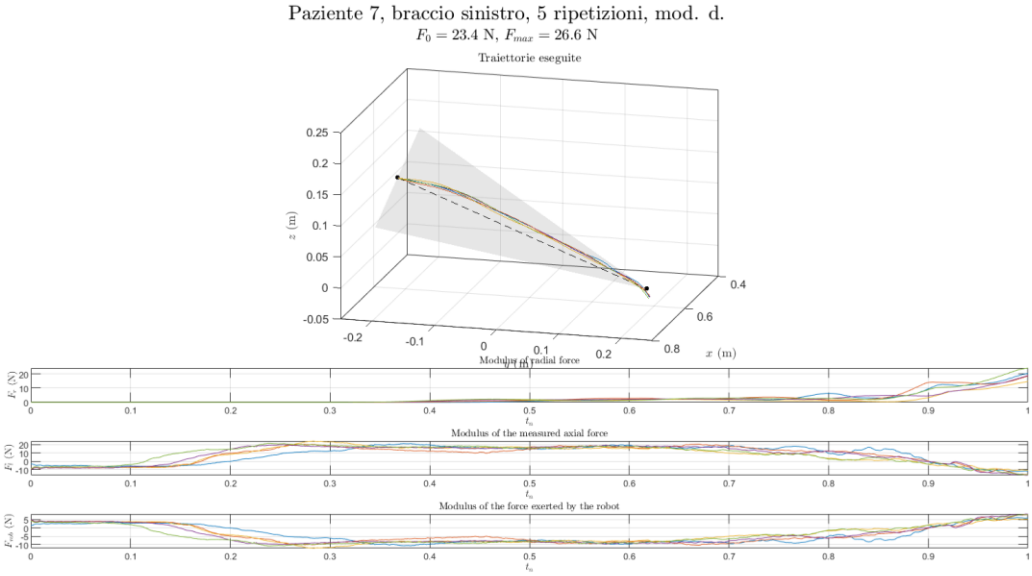


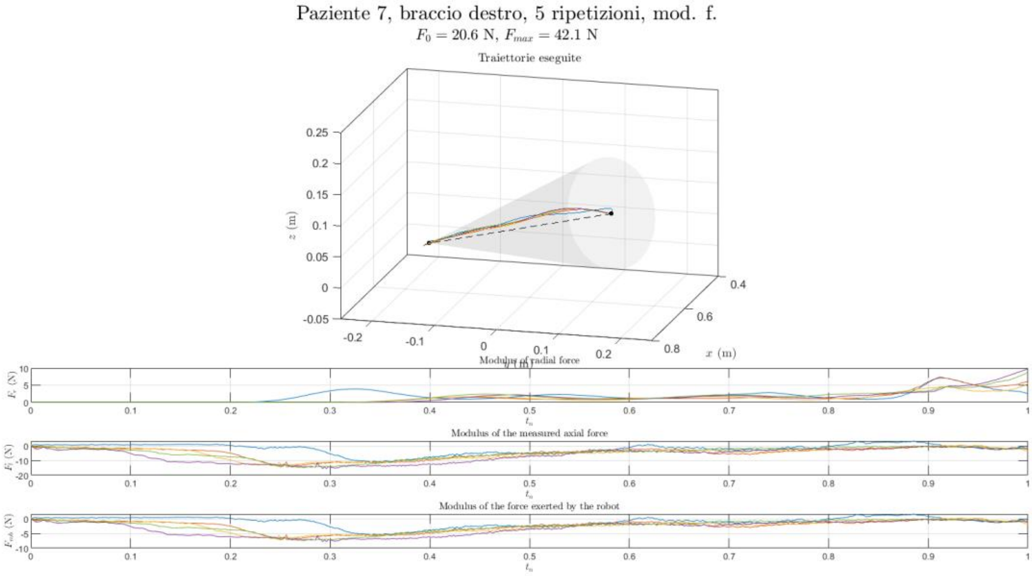
Figure 62 - Force analysis related to healthy patient 16. Lef and right side are here compared

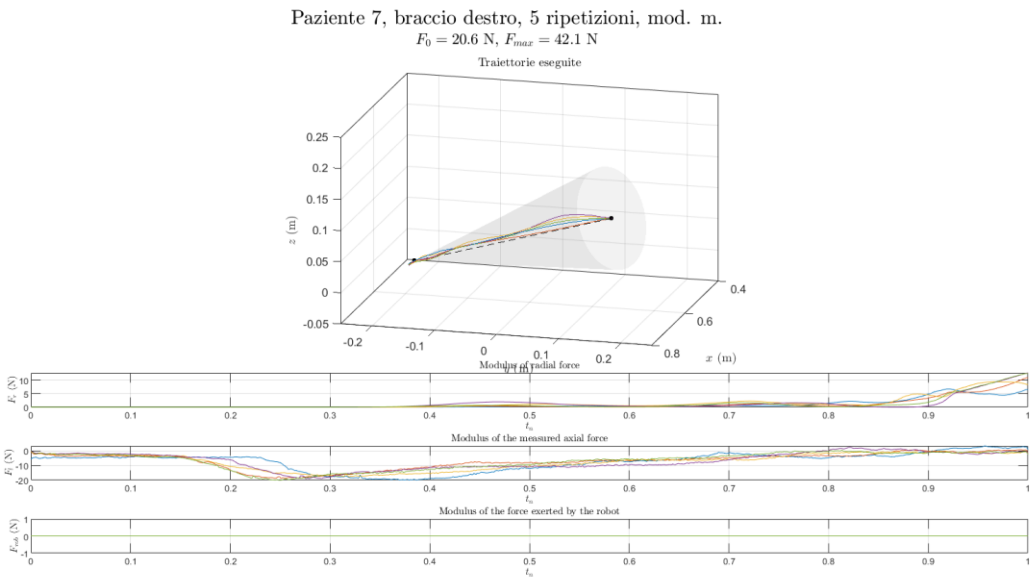
PATIENT 17


Paziente 7, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 23.4$ N, $F_{max} = 26.6$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n


Paziente 7, braccio sinistro, 5 ripetizioni, mod. m.
$F_0 = 23.4$ N, $F_{max} = 26.6$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n


Paziente 7, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 23.4$ N, $F_{max} = 26.6$ N
Traiettorie eseguite
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 7, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 20.6$ N, $F_{max} = 42.1$ N
Traiettorie eseguite
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot



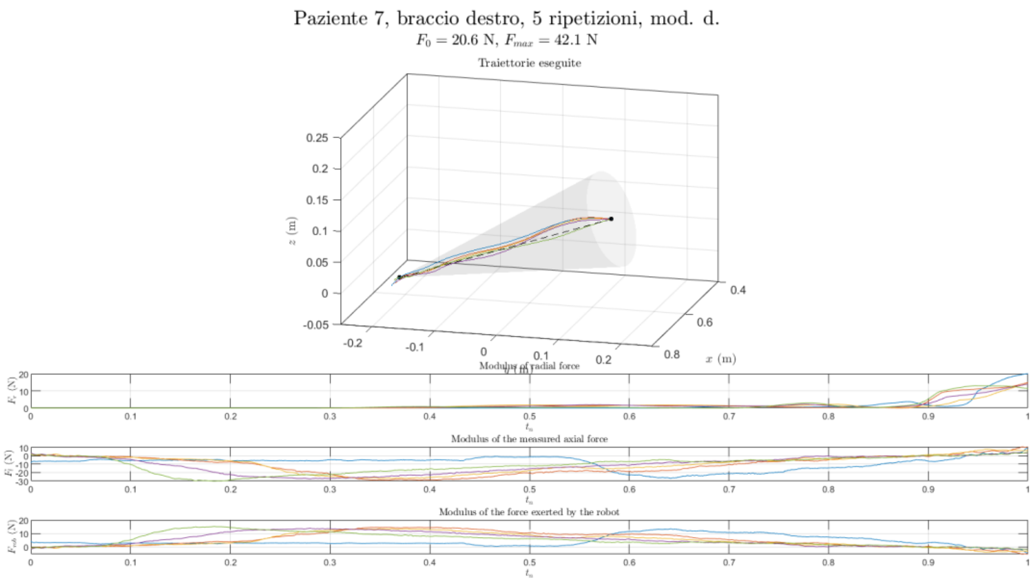
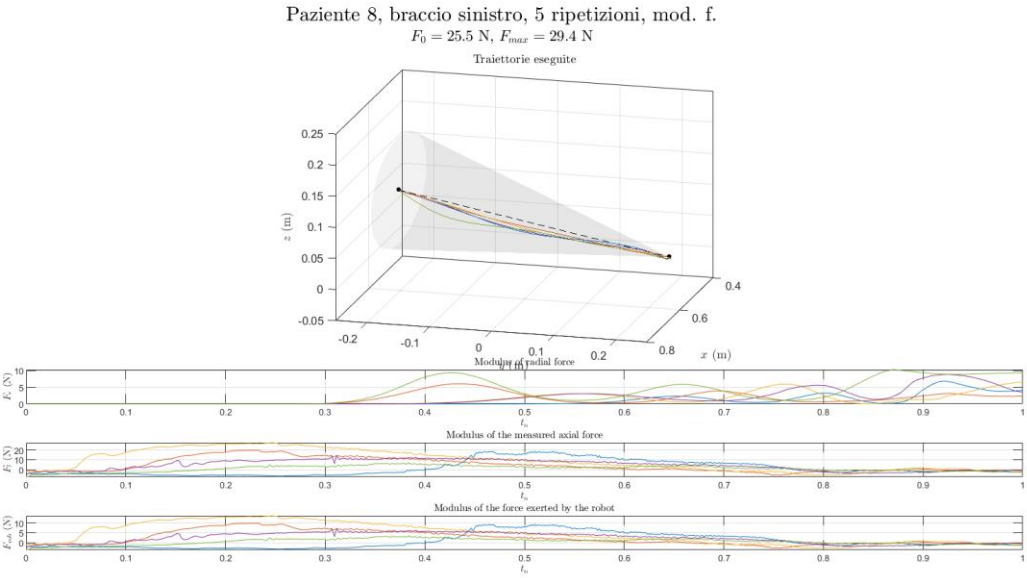
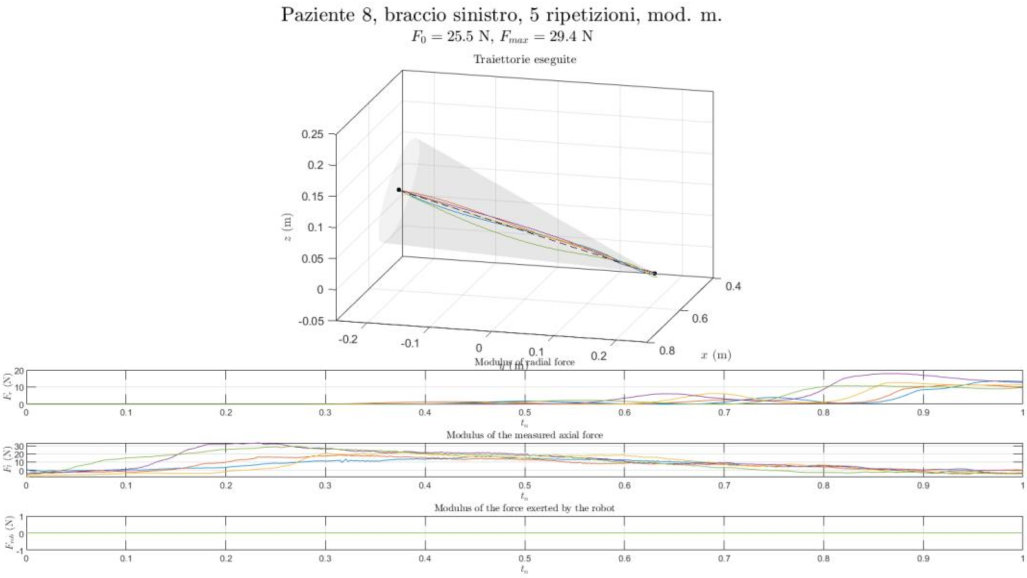


Figure 63 - Force analysis related to healthy patient 17. Lef and right side are here compared.

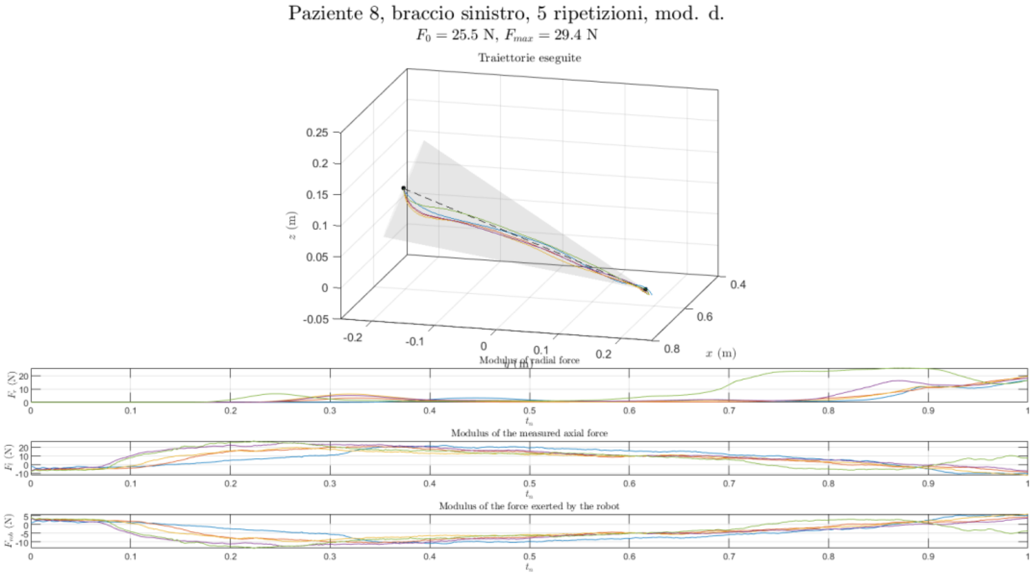
PATIENT 18

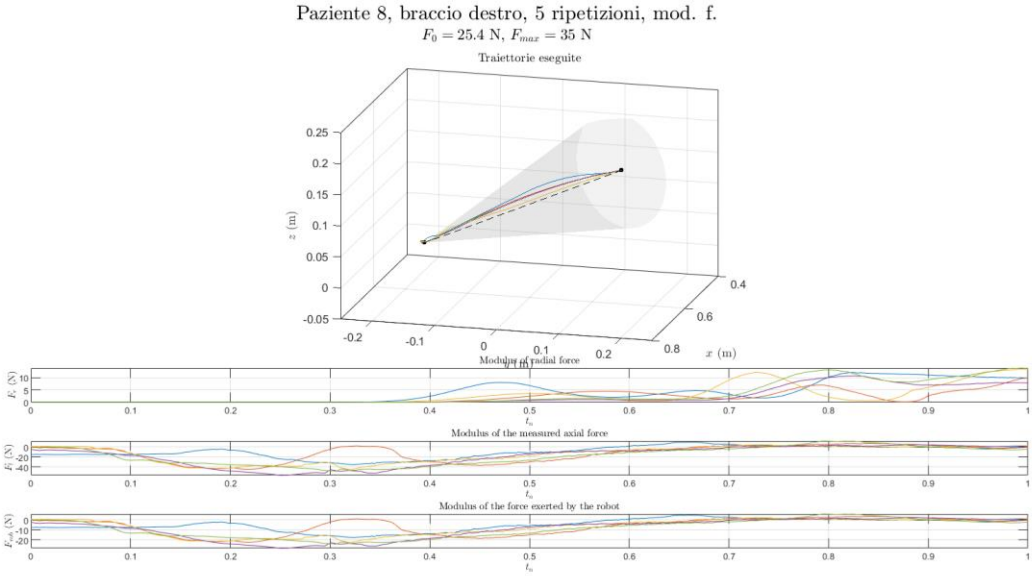


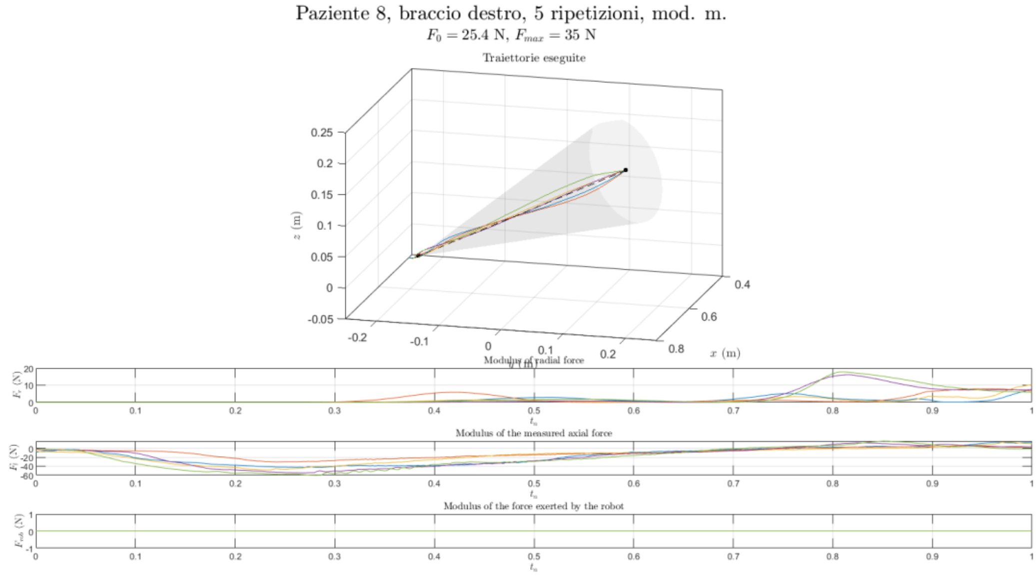
Paziente 8, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 25.5$ N, $F_{max} = 29.4$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot



Paziente 8, braccio sinistro, 5 ripetizioni, mod. m.
$F_0 = 25.5$ N, $F_{max} = 29.4$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 8, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 25.5$ N, $F_{max} = 29.4$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
F_r (N)
Modulus of the measured axial force
F_l (N)
Modulus of the force exerted by the robot
F_{rob} (N)
t_n


Paziente 8, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 25.4$ N, $F_{max} = 35$ N
Traiettorie eseguite
z (m)
x (m)
y (m)
Modulus of radial force
F_r (N)
Modulus of the measured axial force
F_l (N)
Modulus of the force exerted by the robot
F_{rob} (N)
t_n



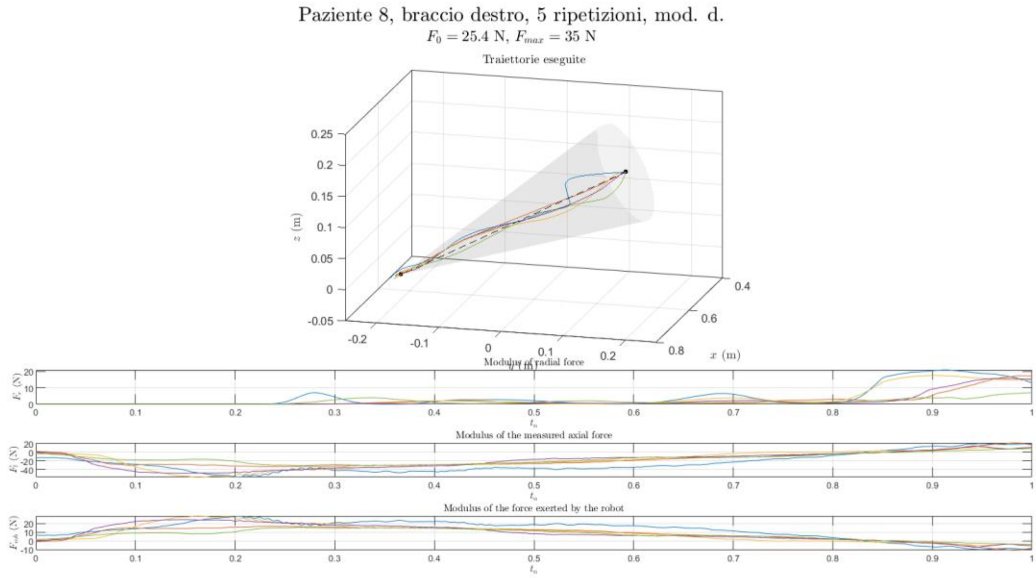
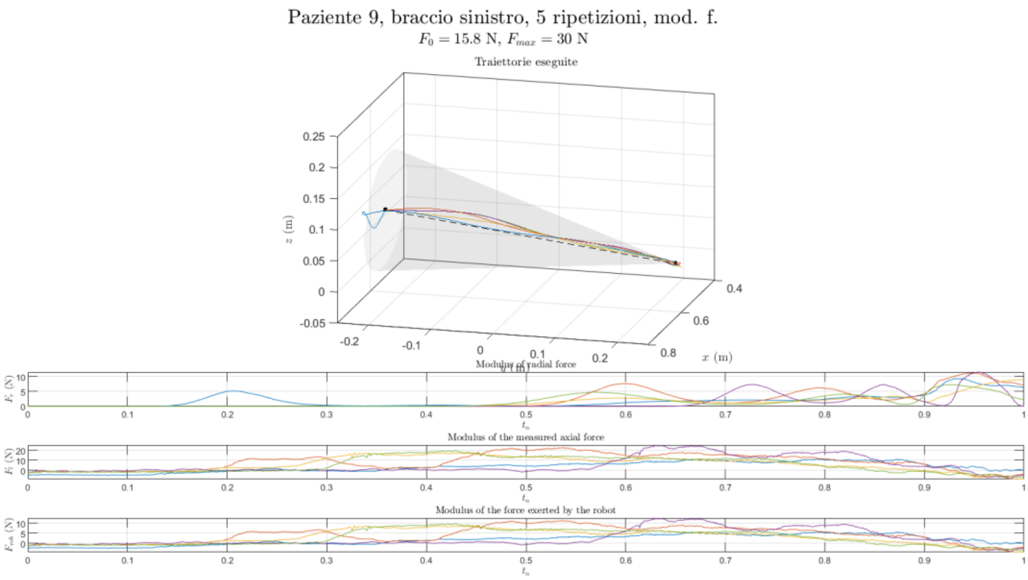
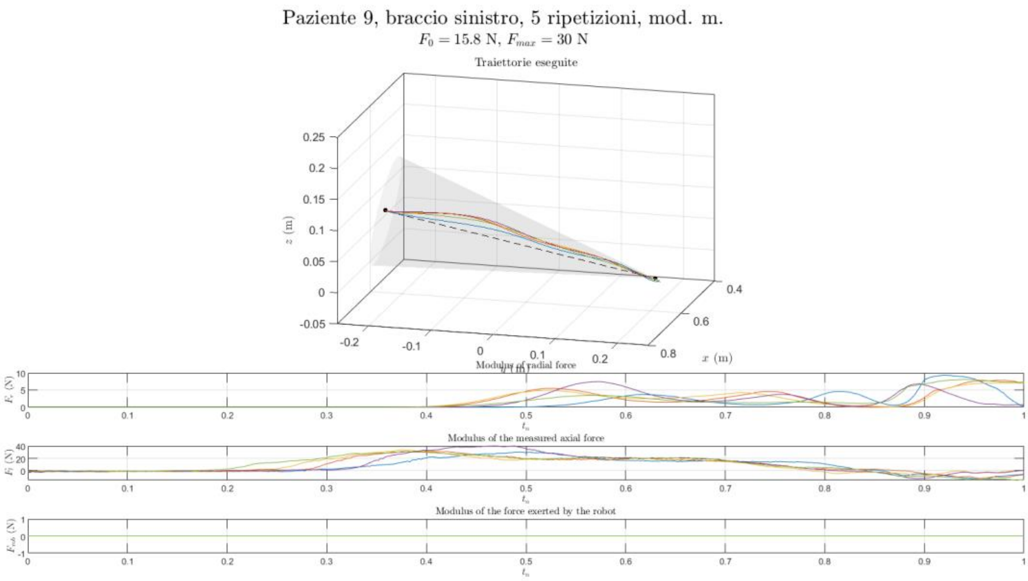


Figure 64 - Force analysis related to healthy patient 18. Lef and right side are here compared

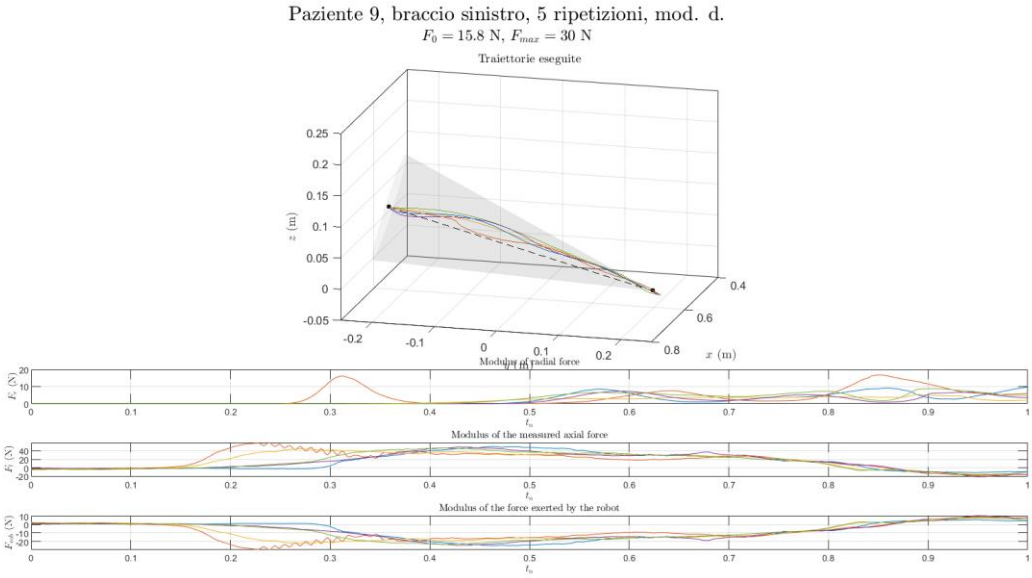
PATIENT 19

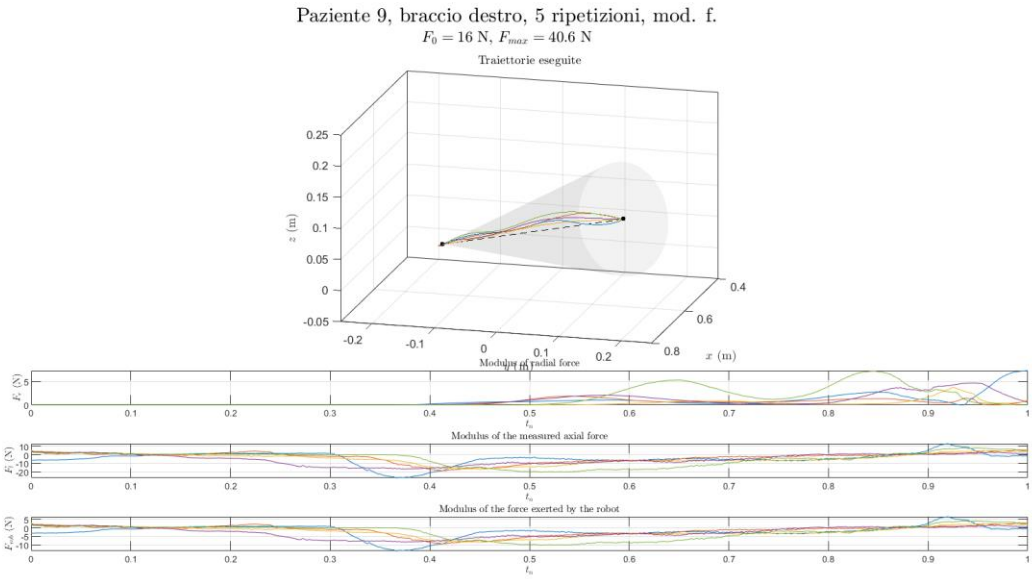


Paziente 9, braccio sinistro, 5 ripetizioni, mod. f.
$F_0 = 15.8$ N, $F_{max} = 30$ N
Traiettorie eseguite
z (m)
y (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n



Paziente 9, braccio sinistro, 5 ripetizioni, mod. m.
$F_0 = 15.8$ N, $F_{max} = 30$ N
Traiettorie eseguite
z (m)
y (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot
t_n


Paziente 9, braccio sinistro, 5 ripetizioni, mod. d.
$F_0 = 15.8$ N, $F_{max} = 30$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot


Paziente 9, braccio destro, 5 ripetizioni, mod. f.
$F_0 = 16$ N, $F_{max} = 40.6$ N
Traiettorie eseguite
z (m)
x (m)
Modulus of radial force
Modulus of the measured axial force
Modulus of the force exerted by the robot

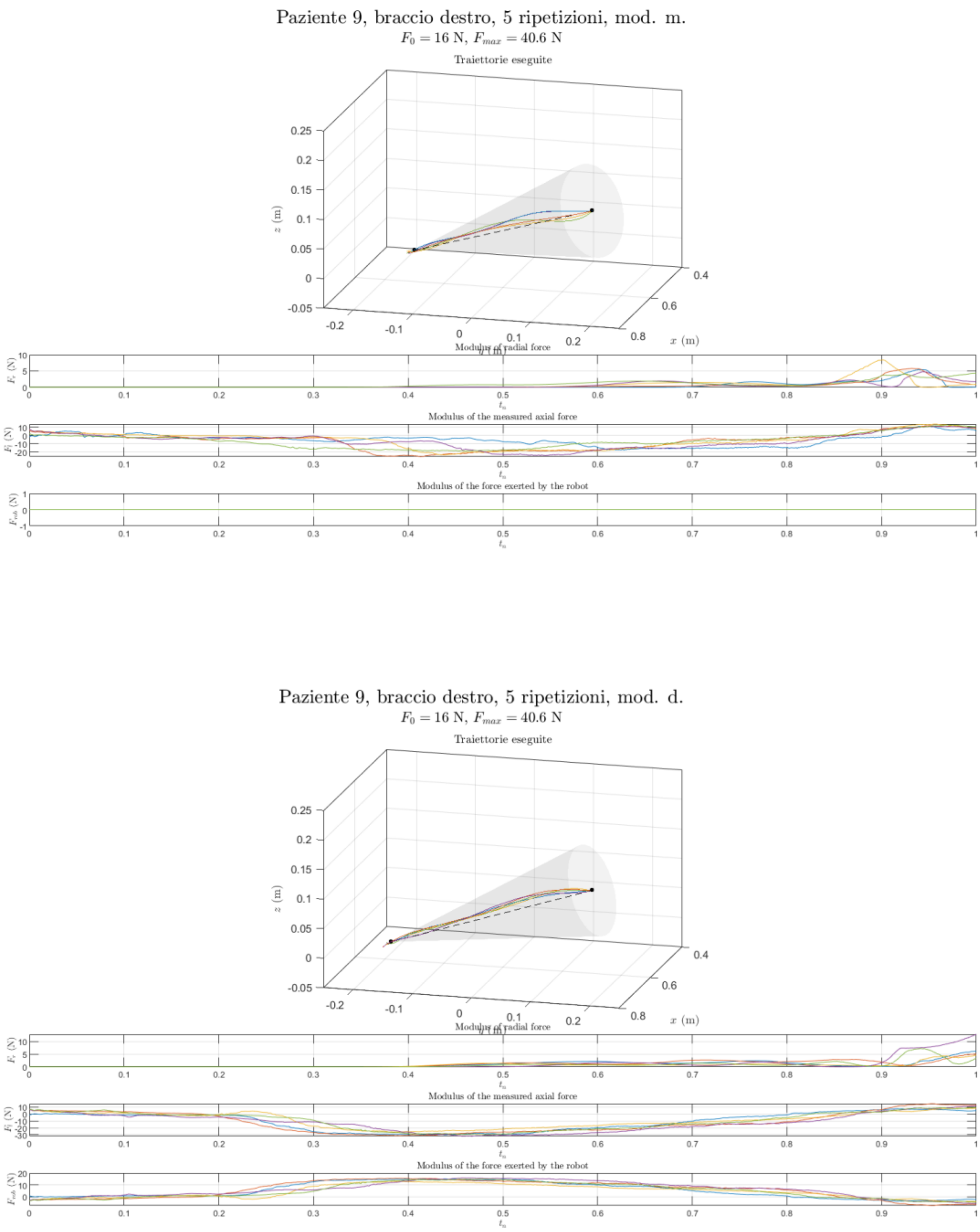


Figure 65 - Force analysis related to healthy patient 19. Lef and right side are here compared.

4.5 Questionnaire results

In Appendix Part B it is reported the questionnaire presented to the patient at the end of the exercise. It was implemented as feedback for the operators. In fact, it asks for feelings and sensations before, during and at the end of the exercise. It was given a point from 1 (not good enough) to 5 (good), and average results are here presented

	Answers n°	Score
Sensations felt AT FIRST SIGHT	1a	4,5
	1b	4,7
Sensation felt DURING the exercise	2a	3,2
	2b	4,9
	2c	4,3
	2d	4,8
	2e	4,5
Ergonomic Handle	3a	4,1
	3b	-
Sensation felt AT THE END of the exercise	4a	4,4
	4b	4,1

CHAPTER 5

CONCLUSIONS

Finally, in this last chapter, conclusions emerging from the previous study are contained, and the further indications for a possible subsequent deepening of the investigation are suggested.

In this preliminary work, the development of a rehabilitation exercise through a cobot UR5e system was presented. The starting point was the focus on the information theoretically required for the use of the robot, by correctly programming it and the evaluation of that in group of patients with upper limbs trauma and in a control group. The aim of this study is to evaluate whether a collaborative robot, such us this type of Universal Robots, could be suitable for an upper limb rehabilitation therapy, for next being integrated in the clinical applications.

Tests performed with patients in this first phase of the project revealed that:

- Patients really appreciated the collaborative robot; moreover, were not worried to work a closely with it;
- Patients would come back with enthusiasm for a future collaboration in the hypothesis to finish a if they had a rehabilitation therapy cycle with the robot;
- Handle grip design was comfortable, even thought some of the patients had problem to grip it related to the clinical history;
- Log Viewer software showed that differences in the curve trends, since pathological ones had unstable and non-uniform trends, while the healthy ones shown regular trends;
- Force analysis in Matlab software was advantageous to visualize the force applied during the exercise: there were big differences between affected side and not affected side. In general, the first ones were characterized with low forces. Moreover, in the control group the values

recorded were constant for both the healthy sides. This last information is a confirmation that the evaluation of forces through Matlab can be a useful tool for post-processing analysis;

All these aspects mentioned above underline the fundamental role of rehabilitation in the recovery of joint stability and in the prevention of relapses but also the need for a thorough and updated knowledge of the scientific evidence by the physiotherapist to formulate a rehabilitation program suitable for those who are innumerable needs and problems that patients with unstable shoulders can have.

6.1 Discussion - Future works

6.1.1 Ergonomic Handle with Microswitch

To perform a rehabilitation exercise in autonomous and continuous way, patients must be first informed and educated of the treatment during the first sessions and robot must be programmed depending on the needs of the patient. Moreover, a button should be implemented. This button play an important role: every time the patient realizes that he has reached the target position, button must be pressed so the robot can start again the exercise loop. The button is composed of a microswitch. A micro switch is a small, very sensitive switch which requires minimum compression to activate, and it operates at very high speed. It normally measures 2 cm x 3 cm, so it is possible to implement in the ergonomic handle. Micro switches generally consist of a Common terminal 'C', Normally Close terminal (NC) and a Normally Open terminal (NO). Microswitches usually have an actuating plunger that only must travel small distance to trigger the contact sequence. A small degree of movement can change the contact positions due to a spring loading mechanism that causes movable contacts to snap between an NO and NC terminals.

The electrical circuit works as it is shown in Figures 66.

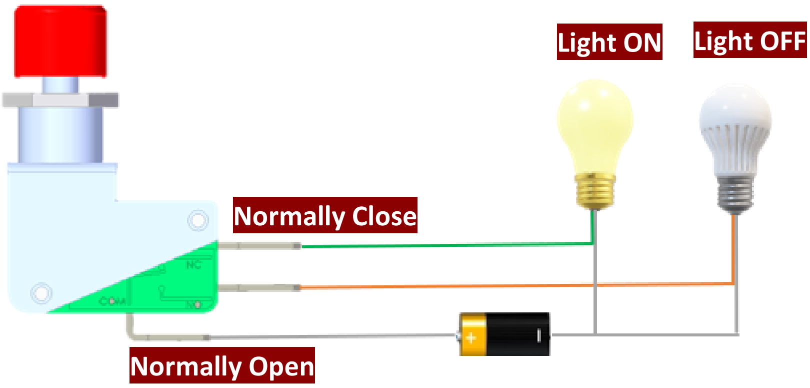


Figure 66 - Electrical circuit of micro switch. When red button will be pressed, the lamp will bright.

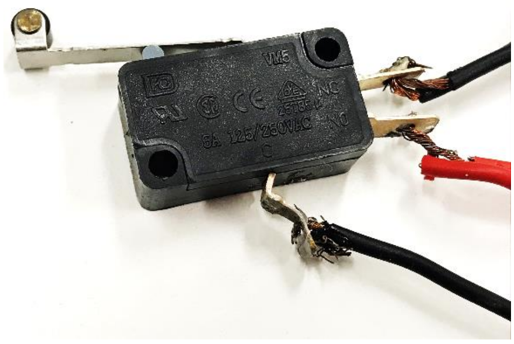


Figure 67 - Microswitch

6.1.2 Modified Ergonomic Handle Design

During experiments, patients had some troubles in holding the handle due to its perpendicular position with respect to the working area. The lesion (stroke in most cases) made the limb of the patients and in particular the hand swollen, not flexible at all and clenched into a fist. Due to these reasons, it is optimal to think in a better design of the handle that considers these problematics, or as was done by the operators during this preliminary study, placed the handle in horizontal layout by helping the patients to hold one finger per time.

APPENDIX

PART A - UR SCRIPT

1st Thread

```
def Primo_Esercizio_Rehab_Fh():
  set_safety_mode_transition_hardness(1)
  set_gravity([0.0, 0.0, 9.82])
  set_standard_analog_input_domain(0, 1)
  set_standard_analog_input_domain(1, 1)
  set_tool_analog_input_domain(0, 1)
  set_tool_analog_input_domain(1, 1)
  set_analog_outputdomain(0, 1)
  set_analog_outputdomain(1, 0)
  set_input_actions_to_default()
  set_tcp(p[0.0,0.0,0.212,0.0,0.0,1.2217])
  set_tool_communication(False, 115200, 0, 1, 1.5, 3.5)
  set_tool_output_mode(0)
  set_tool_digital_output_mode(0, 1)
  set_tool_digital_output_mode(1, 1)
  set_tool_voltage(24)
  step_count_395695b8_8d6e_4732_9f55_e8b006f9c85a = 0.0
  thread Step_Counter_Thread_8fee3aca_3556_4e19_880d_
_e19dd6aaf387():
    while (True):
      step_count_395695b8_8d6e_4732_9f55_e8b006f9c85a =
step_count_395695b8_8d6e_4732_9f55_e8b006f9c85a + 1.0
      sync()
    end
  end
  run
Step_Counter_Thread_8fee3aca_3556_4e19_880d_e19dd6aaf387()
  set_target_payload(0.490000, [0.000000, 0.000000, 0.081000],
[0.000469, 0.000469, 0.000469, 0.000000, 0.000000, 0.000000])
  global cento3=p[-0.4103902373199701,-
0.37800431080170993,0.0495,
-3.1412109469102263,-0.04785991904764895,1.588464559776637E-
5]
  global centro=p[-0.4104395123821702,-
0.2885063861276868,0.0495,3.14114798489791,
0.04789631750033895, -1.9664467121952822E-4]
  global centro1=p[-0.4114171208813015,-
0.2023110442187094,0.0495,
-3.1411372115895353,-0.04773178641621358,-
1.3667440379836558E-4]
  # begin: URCap Installation Node
  #   Source: Cognex In-Sight Robot Guidance, 1.3.1,
 Cognex Corporation
  #   Type: Cognex
  stringRPC = rpc_factory("xmlrpc","http://127.0.0.1:33004")
  socket_open("169.254.0.24", 3105, "CGNX_SOCKET")
  # end: URCap Installation Node
  # begin: URCap Installation Node
  #   Source: SCHUNK Co-act EGP-C for UR, 1.0.0.SNAPSHOT,
SCHUNK GmbH & Co.
  #   Type: SCHUNK Co-act Gripper
  Gripper_LB_Enabled = True
  if (Gripper_LB_Enabled):
    set_digital_out(3, False)
    set_digital_out(2, True)
  end
  def GrpOpen():
  if (Gripper_LB_Enabled):
    set_digital_out(2, False)
    set_digital_out(3, True)
```

```
end
set_tool_digital_out(1, False)
sleep(0.015)
set_tool_digital_out(0, True)
sleep(0.5)
if (Gripper_LB_Enabled):
  set_digital_out(3, False)
  set_digital_out(2, True)
end
end
def GrpClose():
if (Gripper_LB_Enabled):
  set_digital_out(2, False)
  set_digital_out(3, True)
end
set_tool_digital_out(0, False)
sleep(0.015)
set_tool_digital_out(1, True)
sleep(0.5)
if (Gripper_LB_Enabled):
  set_digital_out(3, False)
  set_digital_out(2, True)
end
end
# end: URCap Installation Node
```

2 nd thread

```
global F_h= get_tcp_force ()
global F_r=0
global inizio_prog= False
global inizio_sx=p[0,0,0,0,0,0]
global scelta= False
global timer_1=0
global start_sx_p=p[.645714749702, .000000000000, .242939486107, 2.219424666363, -2.218481389358, .006253140022]
global start_sx_q=[0.20976623205266964, -1.9079211515229222, -1.3311703767250211, -1.4743032739795296, 1.5705144040029007, -1.0113670218327293]
global timer_1_is_counting=False
thread Timer_Thread():
  while (True):
    if (timer_1_is_counting):
      timer_1 = timer_1 + get_steptime()
    end
    sync()
  end
end
run Timer_Thread()
$ 2 "BeforeStart"
$ 3 "zero_ftsensor()"
zero_ftsensor()
$ 4 "ciclo_0≔1"
global ciclo_0=1
$ 142 "Thread_1"
thread Thread_1():
  while (True):
    # begin: URCap Program Node
    #   Source: Cognex In-Sight Robot Guidance, 1.3.1, Cognex Corporation
    #   Type: Cognex Camera Pose
    $ 144 "CGX_result:=CameraPose"
    hasResponse = True
    jobPass = False
    curPose = get_actual_tcp_pose()
    invertVar = False
```

```
socket_send_line(stringRPC.to_xt(curPose), "CGNX_SOCKET")
curResult = socket_read_string("CGNX_SOCKET")
if (curResult):
  invertVar = True
end
if (invertVar):
  hasResponse = False
elif (stringRPC.get_status(curResult)):
  jobPass = True
end
if (hasResponse and jobPass):
  retStr=stringRPC.from_xt(curResult)
  translationPoint = pose_trans(p[0.0,0.0,0.0,0.0,0.0,0.0], p[retStr[0],retStr[1],curPose[2],curPose[3],curPose[4],curPose[5]])
  CGX_result= translationPoint
end
$ 145 "x_target≔CGX_result[0]" "noBreak"
global x_target=CGX_result[0]
$ 146 "y_target≔CGX_result[1]" "noBreak"
global y_target=CGX_result[1]
$ 147 "target_act≔p[x_target,y_target,-0.01,2.22,-2.22,0]" "noBreak"
global target_act=p[x_target,y_target,-0.01,2.22,-2.22,0]
$ 148 "If inizio_prog≟ True " "noBreak"
if (inizio_prog ==   True  ):
  $ 149 "target≔target_act" "noBreak"
  global target=target_act
  $ 150 "inizio_prog≔ False " "noBreak"
  global inizio_prog=  False
end
# begin: URCap Program Node
#   Source: Cognex In-Sight Robot Guidance, 1.3.1, Cognex Corporation
#   Type: Job Pass
$ 151 "Job Pass" "noBreak"
if (hasResponse and jobPass):
end
# end: URCap Program Node
# begin: URCap Program Node
#   Source: Cognex In-Sight Robot Guidance, 1.3.1, Cognex Corporation
#   Type: No Response
$ 157 "No Response" "noBreak"
if not(hasResponse):
  $ 158 "Popup: no response" "noBreak"
  popup("no response", "Message", False, False, blocking=True)
end
# end: URCap Program Node
$ 159 "Wait: 0.01" "noBreak"
sleep(0.01)
# end: URCap Program Node
 end
end
threadId_Thread_1 = run Thread_1()
$ 160 "Thread_2"
thread Thread_2():
 while (True):
  $ 161 "Wait: 0.01"
  sleep(0.01)
  $ 162 "posa_tcp_ini≔get_actual_tcp_pose()"
  global posa_tcp_ini=get_actual_tcp_pose()
  $ 163 "diff≔point_dist(posa_tcp_ini,target)"
  global diff=point_dist(posa_tcp_ini,target)
  $ 164 "If diff<0.01"
  if (diff<0.01):
    $ 165 "Wait: 0.5"
    sleep(0.5)
    $ 166 "Set DO[3]=On"
    set_standard_digital_out(3, True)
    $ 167 "Wait: 0.25"
```

```
    sleep(0.25)
    $ 168 “Set DO[3]=Off”
    set_standard_digital_out(3, False)
    $ 169 “Wait: 1.0”
    sleep(1.0)
  end
```

```
    end
  end
```

3rd thread

```
threadId_Thread_2 = run Thread_2()
 while (True):
  $ 5 “Robot Program”
  $ 6 “Programmi_Script”
  $ 8 “Script: f_robot_dipendente_f_human.script”
  def f_robot_dipendente(f_massima,tipo,target,ori,perc):

  #Calcolo f_uomo
  F_vera = get_tcp_force()

F_vera_Lin=pose_trans(pose_inv(p[target[0],target[1],target[2],ori[0],ori[1],ori[2]]),F_vera)

  #modalita facile: 20% della f_vera forza uomo percepita)
  #modalita media: 50% della f_vera forza uomo percepita)
  #modalita difficile: 90% della f_vera forza uomo percepita)

  if tipo==”f”:
  F_r = F_vera_Lin[2]*perc[0]
  elif tipo==”m”:
  F_r = F_vera_Lin[2]*perc[1]
  elif tipo==”d”:
  F_r = -F_vera_Lin[2]*perc[2]

  end
  return F_r
```

```
  end
  $ 10 “Script: F_molla_distanza.script”
  def f_molla(posa_tcp_ini,ori,target,inizio_sx):
  posa_tcp = get_actual_tcp_pose()

  #Calcolo  di K in funzione della distanza

  # distanza massima e raggio massimo
  d_max = point_dist(posa_tcp,target)
  r_max = d_max/5

  d_ori =
pose_trans(pose_inv(p[target[0],target[1],target[2],ori[0],ori[1],ori[2]]),get_actual_tcp_pose())
  d = sqrt(d_ori[2]*d_ori[2])
  r = sqrt(d_ori[0]*d_ori[0]+d_ori[1]*d_ori[1])

  m = d_max/r_max #coeff angolare della retta d_ori_0 = m * v

  r_0 = (1/m) * d

  #cubica per condizione r<r_0

  k_f = 2000  #N/m
```

```
#Calcolo
if r < r_0:
k =  ((-k_f*r*r)/(r_0*r_0*r_0))*(2*r-3*r_0)
else:
k = k_f
end

#Calcolo la Forza della molla in funzione di K funzione della distanza tra il target ed il punto iniziale
F_r_opposta_x = -k * d_ori[0]
F_r_opposta_y = -k * d_ori[1]
F_r_opposta_z = -k * d_ori[2]*0

return [F_r_opposta_x,F_r_opposta_y,F_r_opposta_z]
#return [F_r_opposta_x,F_r_opposta_y,F_r_opposta_z,z_x,z_y,z_z]
end
$ 11 “Script: Cambio_Riferimento.script”
def change_orientation(pstart,target):

#pstart = punto iniziale
x_i = pstart[0]
y_i = pstart[1]
z_i = pstart[2]

#target = punto finale
x_f = target[0]
y_f = target[1]
z_f = target[2]

#vettore s
s_x = (x_f-x_i)*(1/sqrt(pow(x_f-x_i,2)+pow(y_f-y_i,2)+pow(z_f-z_i,2)))
s_y = (y_f-y_i)*(1/sqrt(pow(x_f-x_i,2)+pow(y_f-y_i,2)+pow(z_f-z_i,2)))
s_z = (z_f-z_i)*(1/sqrt(pow(x_f-x_i,2)+pow(y_f-y_i,2)+pow(z_f-z_i,2)))

#Parameters

u_x = (sqrt((4*s_y*sqrt(s_x*s_x + s_y*s_y) - s_x*s_x*s_z + 3*s_x*s_x + 4*s_y*s_y)/(s_x*s_x*(s_z + 1)))*(4*s_x*s_x*s_y - 4*s_y*s_y*sqrt(s_x*s_x + s_y*s_y) - 3*s_x*s_x*sqrt(s_x*s_x + s_y*s_y) + 4*s_y*s_y*s_y + s_x*s_x*s_z*sqrt(s_x*s_x + s_y*s_y)))/(s_x*s_x*s_z*s_z - 6*s_x*s_x*s_z + 9*s_x*s_x - 8*s_y*s_y*s_z + 8*s_y*s_y)

u_y = (s_x*sqrt((4*s_y*sqrt(s_x*s_x + s_y*s_y) - s_x*s_x*s_z + 3*s_x*s_x + 4*s_y*s_y)/(s_x*s_x*(s_z + 1)))*(4*s_y*sqrt(s_x*s_x + s_y*s_y) + 4*s_x*s_x*s_z + 4*s_y*s_y*s_z - 12*s_x*s_x - 12*s_y*s_y + 4*s_y*s_z*sqrt(s_x*s_x + s_y*s_y)))/(4*s_x*s_x*s_z*s_z - 24*s_x*s_x*s_z + 36*s_x*s_x - 32*s_y*s_y*s_z + 32*s_y*s_y)

u_z = (s_x*(s_z + 1)*sqrt((4*s_y*sqrt(s_x*s_x + s_y*s_y) - s_x*s_x*s_z + 3*s_x*s_x + 4*s_y*s_y)/(s_x*s_x*(s_z + 1)))*(s_y + s_z*sqrt(s_x*s_x + s_y*s_y) + s_y*s_z - 3*sqrt(s_x*s_x + s_y*s_y)))/(s_x*s_x*s_z*s_z - 6*s_x*s_x*s_z + 9*s_x*s_x - 8*s_y*s_y*s_z + 8*s_y*s_y)

theta = -2*atan(sqrt((4*s_y*sqrt(s_x*s_x + s_y*s_y) - s_x*s_x*s_z + 3*s_x*s_x + 4*s_y*s_y)/(s_x*s_x*(s_z + 1))))

rx = theta*u_x
```

```
    ry = theta*u_y
    rz = theta*u_z

    return[rx,ry,rz]

  end
  $ 12 "target≔p[0,0,0,0,0,0]"
  global target=p[0,0,0,0,0,0]
  $ 13 "Braccio DX o SX"
  $ 14 "braccio≔'Braccio Destro o Sinistro? Destro = d; Sinistro = s'"
  global braccio=request_string_from_primary_client("Braccio Destro o Sinistro? Destro = d; Sinistro = s")
  $ 15 "If braccio≟'d'"
  if (braccio == "d"):
    $ 16 "start≔p[0.693,0.430,0.100,1.42,-2.8,0]"
    global start=p[0.693,0.430,0.100,1.42,-2.8,0]
  else:
    $ 17 "ElseIf braccio≟'s'"
    if (braccio == "s"):
      $ 18 "start≔p[0.693,-0.430,0.100,1.42,-2.8,0]"
      global start=p[0.693,-0.430,0.100,1.42,-2.8,0]
    end
  end
  $ 19 "MoveL"
  $ 21 "start" "breakAfter"
  movel(start, a=0.8, v=0.1)
  $ 22 "Setup Force mode"
  $ 24 "tipo≔'Modalita' di esercizio? Facile = f  medio = m  difficile = d'"
  global tipo=request_string_from_primary_client("Modalita' di esercizio? Facile = f  medio = m  difficile = d")
  $ 25 "perc≔[0.5,0,0.5]"
  global perc=[0.5,0,0.5]
  $ 26 "If tipo≟'f'"
  if (tipo == "f"):
    $ 27 "damp_esercizio≔0.001"
    global damp_esercizio=0.001
  else:
    $ 29 "ElseIf tipo≟'m'"
    if (tipo == "m"):
      $ 31 "damp_esercizio≔0.001"
      global damp_esercizio=0.001
    else:
      $ 32 "ElseIf tipo≟'d'"
      if (tipo == "d"):
        $ 34 "damp_esercizio≔0.001"
        global damp_esercizio=0.001
      end
    end
  end
  $ 35 "Loop inizio_prog≟ False "
  while (inizio_prog ==   False  ):
    $ 36 "inizio_prog≔'Posizionare il target nell'area di lavoro. Fatto?'"
    global inizio_prog=request_boolean_from_primary_client("Posizionare il target nell'area di lavoro. Fatto?")
  end
  $ 37 "ripetizione≔'Quante volte ripetere l'esercizio?'"
  global ripetizione=request_float_from_primary_client("Quante volte ripetere l'esercizio?")
  $ 38 "If ciclo_0≟1"
  if (ciclo_0 == 1):
    $ 39 "start_sx" "breakAfter"
    movel(start_sx_p, a=0.8, v=0.1)
    $ 40 "freedrive_mode(freeAxes=[1,1,1,0,0,0],feature = p[0,0,0,0,0,0])"
    freedrive_mode(freeAxes=[1,1,1,0,0,0],feature = p[0,0,0,0,0,0])
```

```
$ 41 “scelta≔’Set up completato?’”
global scelta=request_boolean_from_primary_client(“Set up completato?”)
$ 42 “If scelta≟ True “
if (scelta ==   True  ):
  $ 43 “end_freedrive_mode()”
  end_freedrive_mode()
end
$ 44 “inizio_sx≔get_actual_tcp_pose()”
global inizio_sx= get_actual_tcp_pose ()
$ 45 “inizio_sx” “breakAfter”
movel(inizio_sx, a=0.8, v=0.1)
$ 46 “timer_1: Start”
timer_1_is_counting = True
$ 47 “calib_f≔0”
global calib_f=0
$ 48 “Suono_1”
$ 49 “Set DO[3]=On”
set_standard_digital_out(3, True)
$ 50 “Wait: 0.1”
sleep(0.1)
$ 51 “Set DO[3]=Off”
set_standard_digital_out(3, False)
$ 52 “Wait: 1.0”
sleep(1.0)
$ 53 “n_misure≔0”
global n_misure=0
$ 54 “Procedura Peso braccio”
$ 55 “Wait: 3.0”
sleep(3.0)
$ 56 “Loop timer_1<5”
while (timer_1<5):
  $ 57 “calib_f≔calib_f+force()”
  global calib_f=calib_f+force()
  $ 58 “n_misure≔n_misure+1”
  global n_misure=n_misure+1
  $ 59 “Wait: 0.1”
  sleep(0.1)
end
$ 60 “calib_f≔(calib_f/n_misure)*1.2”
global calib_f=(calib_f/n_misure)*1.2
$ 61 “timer_1: Stop”
timer_1_is_counting = False
$ 62 “timer_1≔0”
global timer_1=0
$ 63 “Suono_1”
$ 64 “Set DO[3]=On”
set_standard_digital_out(3, True)
$ 65 “Wait: 0.1”
sleep(0.1)
$ 66 “Set DO[3]=Off”
set_standard_digital_out(3, False)
$ 67 “Wait: 1.0”
sleep(1.0)
$ 68 “timer_1: Start”
timer_1_is_counting = True
$ 69 “f_massima≔0”
global f_massima=0
$ 70 “Procedura per la forza massima del paziente”
$ 71 “Wait: 3.0”
sleep(3.0)
$ 72 “Loop timer_1<3”
while (timer_1<3):
  $ 73 “f_act≔force()”
  global f_act=force()
  $ 74 “If f_act>f_massima”
```

```
    if (f_act>f_massima):
      $ 75 "f_massima≔f_act"
      global f_massima=f_act
    end
    $ 76 "Wait: 0.1"
    sleep(0.1)
  end
  $ 77 "timer_1: Stop"
  timer_1_is_counting = False
  $ 78 "timer_1≔0"
  global timer_1=0
  $ 79 "ciclo_0≔0"
  global ciclo_0=0
end
$ 80 "MoveL"
$ 81 "inizio_sx" "breakAfter"
movel(inizio_sx, a=0.8, v=0.1)
$ 82 "Suono_2"
$ 83 "Set DO[3]=On"
set_standard_digital_out(3, True)
$ 84 "Wait: 0.25"
sleep(0.25)
$ 85 "Set DO[3]=Off"
set_standard_digital_out(3, False)
$ 86 "Wait: 0.25"
sleep(0.25)
$ 87 "Set DO[3]=On"
set_standard_digital_out(3, True)
$ 88 "Wait: 0.25"
sleep(0.25)
$ 89 "Set DO[3]=Off"
set_standard_digital_out(3, False)
$ 90 "Wait: 0.25"
sleep(0.25)
$ 91 "esercizio_on≔0"
global esercizio_on=0
$ 92 "Loop ripetizione ≠0"
while (ripetizione  != 0):
  $ 93 "Loop digital_out[3]≠ True "
  while (get_standard_digital_out(3) !=   True  ):
    $ 94 "ori≔change_orientation(inizio_sx,target)"
    global ori=change_orientation(inizio_sx,target)
    $ 95 "F_h≔get_tcp_force()"
    global F_h=get_tcp_force()
    $ 96
"F_h_Lin≔pose_trans(pose_inv(p[target[0],target[1],target[2],ori[
0],ori[1],ori[2]]),F_h)"
    global
F_h_Lin=pose_trans(pose_inv(p[target[0],target[1],target[2],ori[0],
ori[1],ori[2]]),F_h)
    $ 97 "F_h_LinY≔F_h_Lin[2]"
    global F_h_LinY=F_h_Lin[2]
    $ 98
"F_r≔f_robot_dipendente(f_massima,tipo,target,ori,perc)"
    global F_r=f_robot_dipendente(f_massima,tipo,target,ori,perc)
    $ 99 "F_r_molla≔f_molla(posa_tcp_ini,ori,target,inizio_sx)"
    global F_r_molla=f_molla(posa_tcp_ini,ori,target,inizio_sx)
    $ 100 "ori≔change_orientation(inizio_sx,target)"
    global ori=change_orientation(inizio_sx,target)
    $ 101
"force_mode(p[target[0],target[1],target[2],ori[0],ori[1],ori[2]],[1,1,
1,0,0,0],[F_r_molla[0],F_r_molla[1],F_r,0,0,0],2,[1,1,1,0.785,0.785,0.
785])"

force_mode(p[target[0],target[1],target[2],ori[0],ori[1],ori[2]],[1,1,
1,0,0,0],[F_r_molla[0],F_r_molla[1],F_r,0,0,0],2,[1,1,1,0.785,0.785,0.
785])
    $ 102 "If (force()>calib_f)"
    if ((force()>calib_f)):
      $ 103 "force_mode_set_damping(damp_esercizio)"
```

```
      force_mode_set_damping(damp_esercizio)
      $ 104 "esercizio_on≔1"
      global esercizio_on=1
    else:
      $ 105 "ElseIf esercizio_on≟0"
      if (esercizio_on == 0):
        $ 106 "force_mode_set_damping(1)"
        force_mode_set_damping(1)
        $ 107
"force_mode(p[target[0],target[1],target[2],ori[0],ori[1],ori[2]],[0,0,
0,0,0,0],[F_r_molla[0],F_r_molla[1],F_r,0,0,0],2,[1,1,1,0.785,0.785,0.
785])"

force_mode(p[target[0],target[1],target[2],ori[0],ori[1],ori[2]],[0,0,
0,0,0,0],[F_r_molla[0],F_r_molla[1],F_r,0,0,0],2,[1,1,1,0.785,0.785,0.
785])
      end
    end
    $ 108 "sync()"
    sync()
  end
  $ 110 "end_force_mode()"
  end_force_mode()
  $ 111 "esercizio_on≔0"
  global esercizio_on=0
  $ 112 "ripetizione≔ripetizione-1"
  global ripetizione=ripetizione-1
  $ 113 "If ripetizione≠0"
  if (ripetizione != 0):
    $ 114 "MoveL"
    $ 115 "inizio_sx" "breakAfter"
    movel(inizio_sx, a=0.8, v=0.1)
    $ 116 "Suono_2"
    $ 117 "Set DO[3]=On"
    set_standard_digital_out(3, True)
    $ 118 "Wait: 0.25"
    sleep(0.25)
    $ 119 "Set DO[3]=Off"
    set_standard_digital_out(3, False)
    $ 120 "Wait: 0.25"
    sleep(0.25)
    $ 121 "Set DO[3]=On"
    set_standard_digital_out(3, True)
    $ 122 "Wait: 0.25"
    sleep(0.25)
    $ 123 "Set DO[3]=Off"
    set_standard_digital_out(3, False)
    $ 124 "Wait: 0.25"
    sleep(0.25)
  else:
    $ 125 "Else" "noBreak"
    $ 126
"p_alto≔p[target[0],target[1],0.03,target[3],target[4],target[5]]"
    global
p_alto=p[target[0],target[1],0.03,target[3],target[4],target[5]]
    $ 127 "p_alto" "breakAfter"
    movel(p_alto, a=0.8, v=0.1)
    $ 129 "start" "breakAfter"
    movel(start, a=0.8, v=0.1)
    $ 130 "Suono_3"
    $ 131 "Set DO[3]=On"
    set_standard_digital_out(3, True)
    $ 132 "Wait: 0.25"
    sleep(0.25)
    $ 133 "Set DO[3]=Off"
    set_standard_digital_out(3, False)
    $ 134 "Wait: 0.25"
    sleep(0.25)
    $ 135 "Set DO[3]=On"
```

```
set_standard_digital_out(3, True)
$ 136 "Wait: 0.25"
sleep(0.25)
$ 137 "Set DO[3]=Off"
set_standard_digital_out(3, False)
$ 138 "Wait: 0.25"
sleep(0.25)
$ 139 "Set DO[3]=On"
set_standard_digital_out(3, True)
    $ 140 "Wait: 0.25"
    sleep(0.25)
    $ 141 "Set DO[3]=Off"
    set_standard_digital_out(3, False)
   end
  end
 end
end
```

PART B - Questionnarie Form

Età	
Sesso	
Peso	
Braccio Dominante	

____________________________________QUESTIONARIO____________________________________

1. **Sensazioni provate PRIMA dell'esercizio**

   a. Che impressioni hai avuto appena hai visto il robot?
   0: diffidenza e paura del contatto, 5: curiosità e voglia di provare

0	1	2	3	4	5

   b. Sei stato sufficiente informato sulla metodica dell'esercizio prima di svolgerlo?
   0: decisamente no, 5: decisamente si

0	1	2	3	4	5

2. **Sensazioni provate DURANTE l'esercizio:**

   a. Quanto è stato difficile per te eseguire l'esercizio?
   0: molto difficile, 5: molto facile

0	1	2	3	4	5

   b. Hai provato paura e/o preoccupazione durante l'esercizio? (Ad esempio: paura che il robot possa far male all'arto):
   0: decisamente si, 5: decisamente no

0	1	2	3	4	5

   c. Quanto ti sei divertito nello svolgimento dell'esercizio?
   0: per niente divertito, 5: tantissimo divertito

0	1	2	3	4	5

   d. Quanto ti sei annoiato nello svolgimento dell'esercizio?
   0: molto annoiato, 5: per niente annoiato

0	1	2	3	4	5

e. Quanto sei soddisfatto dell'esercizio eseguito?
0: decisamente no, 5: decisamente si

0	1	2	3	4	5

3. Maniglia:

a. Comodità della maniglia (grigia)
0: scomoda, 5: comoda

0	1	2	3	4	5

b. Comodità del guanto (nero)
0: scomoda, 5: comoda

0	1	2	3	4	5

4. Sensazioni provate dopo l'esercizio:

a. Ti sentiresti di consigliarlo a qualcun altro?
0: decisamente no, 5: decisamente si

0	1	2	3	4	5

b. Torneresti un'altra volta per continuare il ciclo di riabilitazione del braccio?
0: decisamente no, 5: decisamente si

0	1	2	3	4	5

PART C- Log Viewer Analysis

PATHOLOGICAL SUBJECT 3

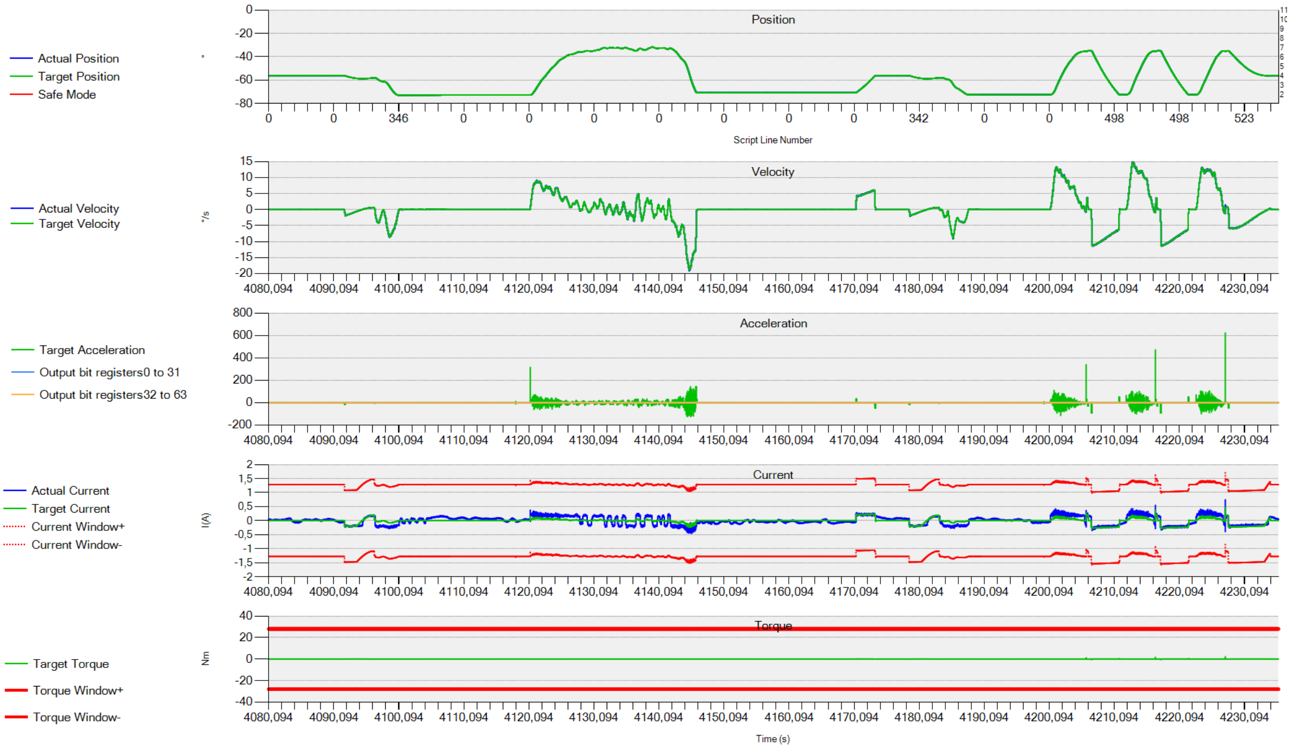


Figure 68 - Pathological subject 3. Left Side. F-Modality

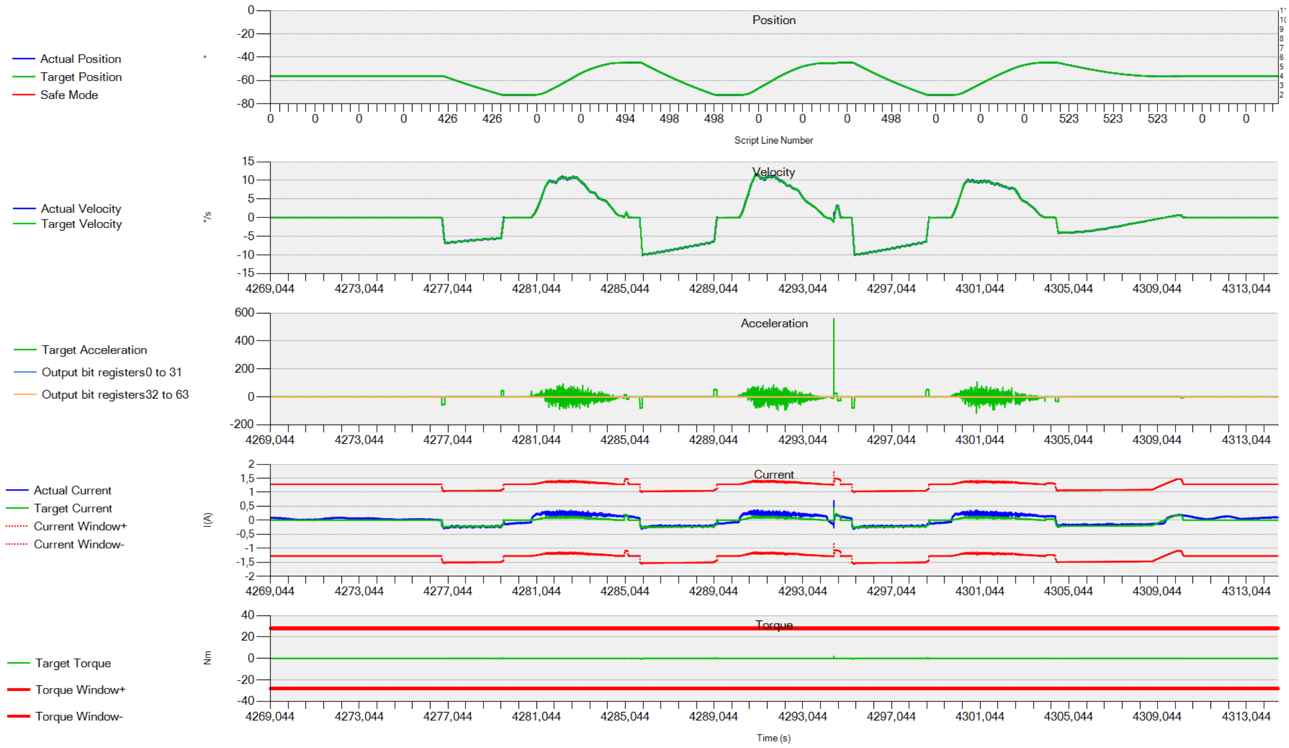


Figure 69 - Pathological subject 3. Left Side. M-Modality

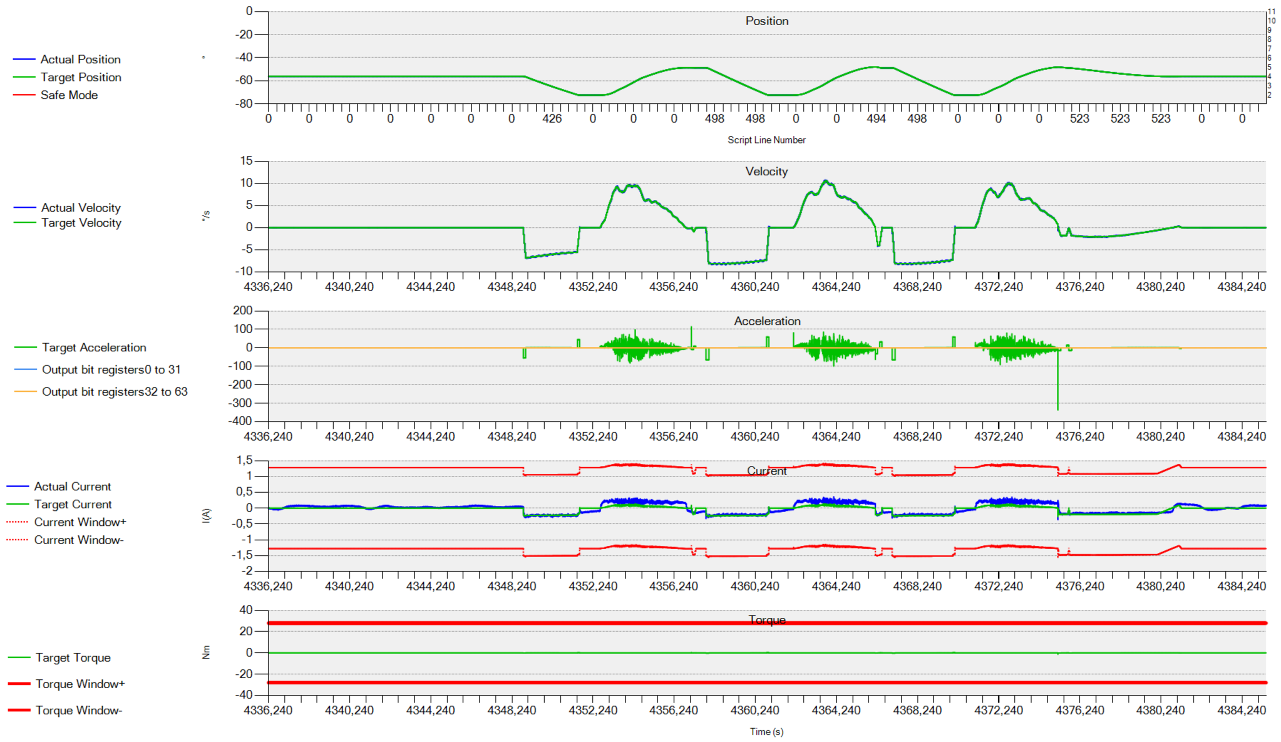


Figure 70 - Pathological subject 3. Left Side. D-Modality

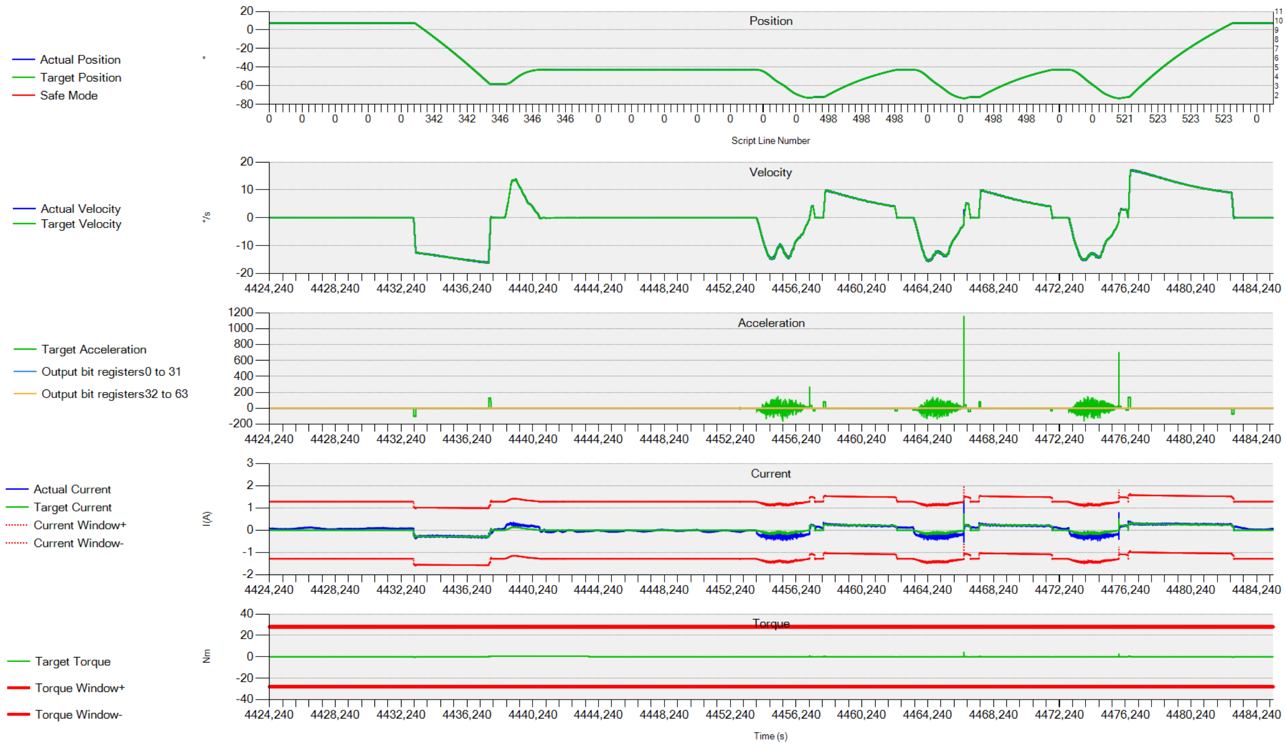


Figure 71- Pathological subject 3. Right Side. F-Modality

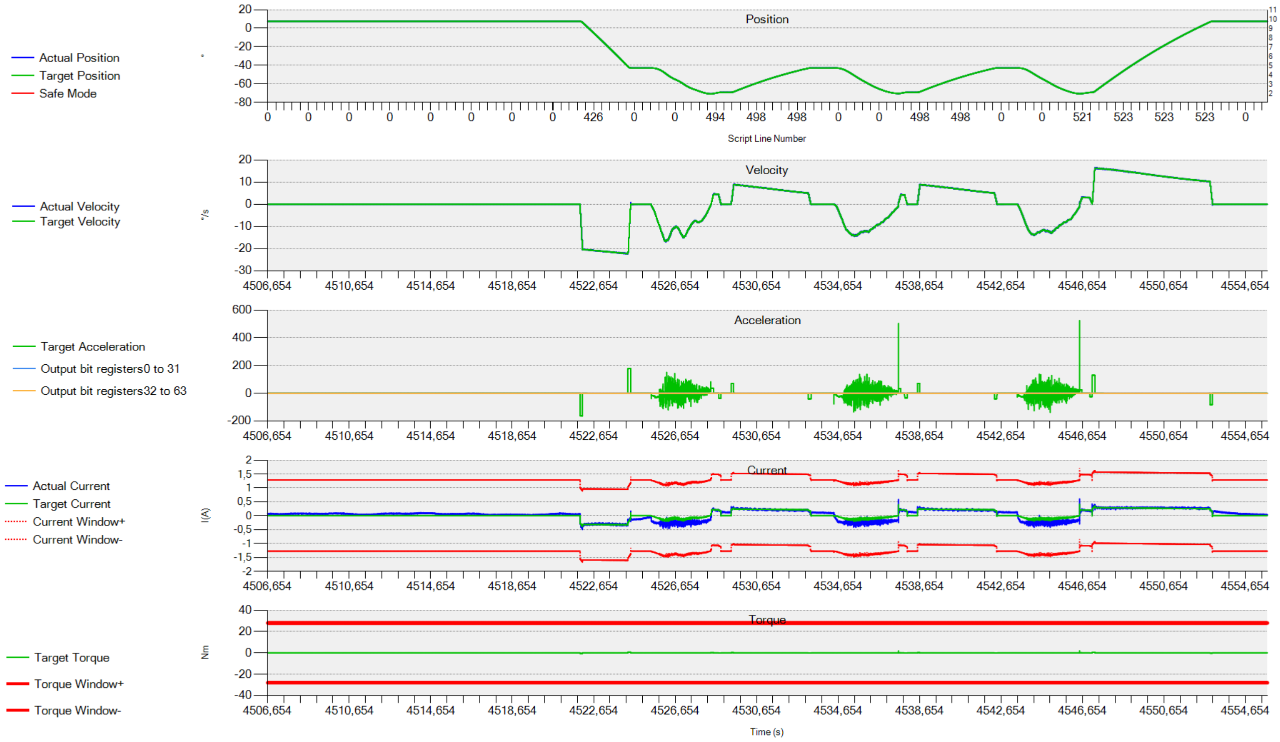


Figure 72 - Pathological subject 3. Right Side. M-Modality.

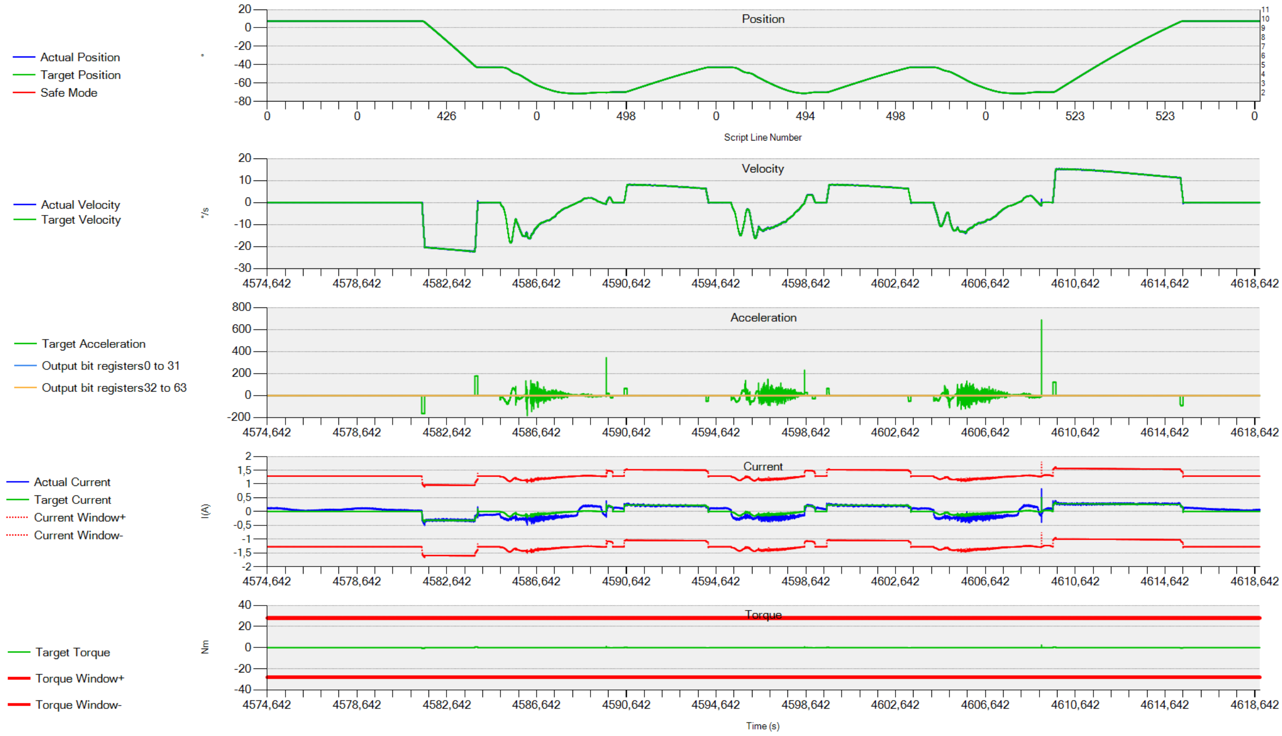


Figure 73- Pathological subject 3. Right Side. D-Modality

PATHOLOGICAL SUBJECT 4

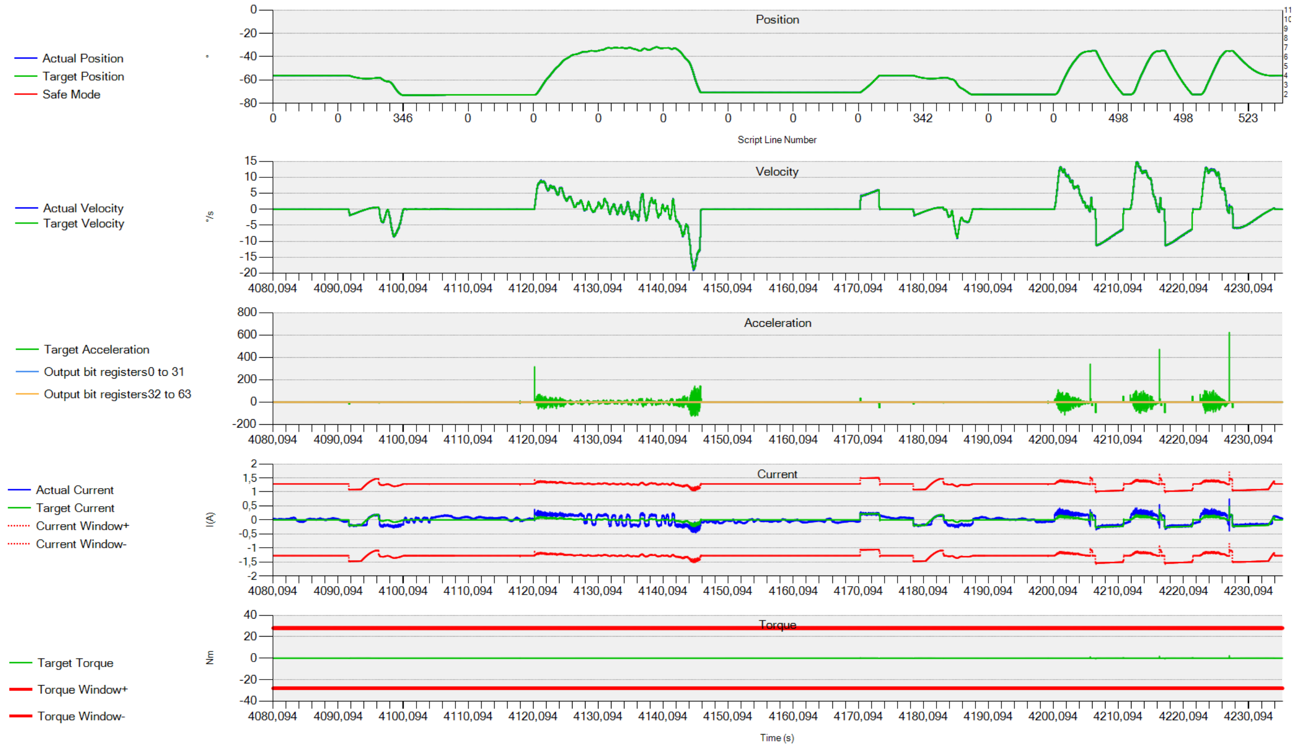


Figure 74 - Pathological subject 4. Left Side. F-Modality

Figure 75 - Pathological subject 4. Left Side. M-Modality

Figure 76 - Pathological subject 4. Left Side. D-Modality

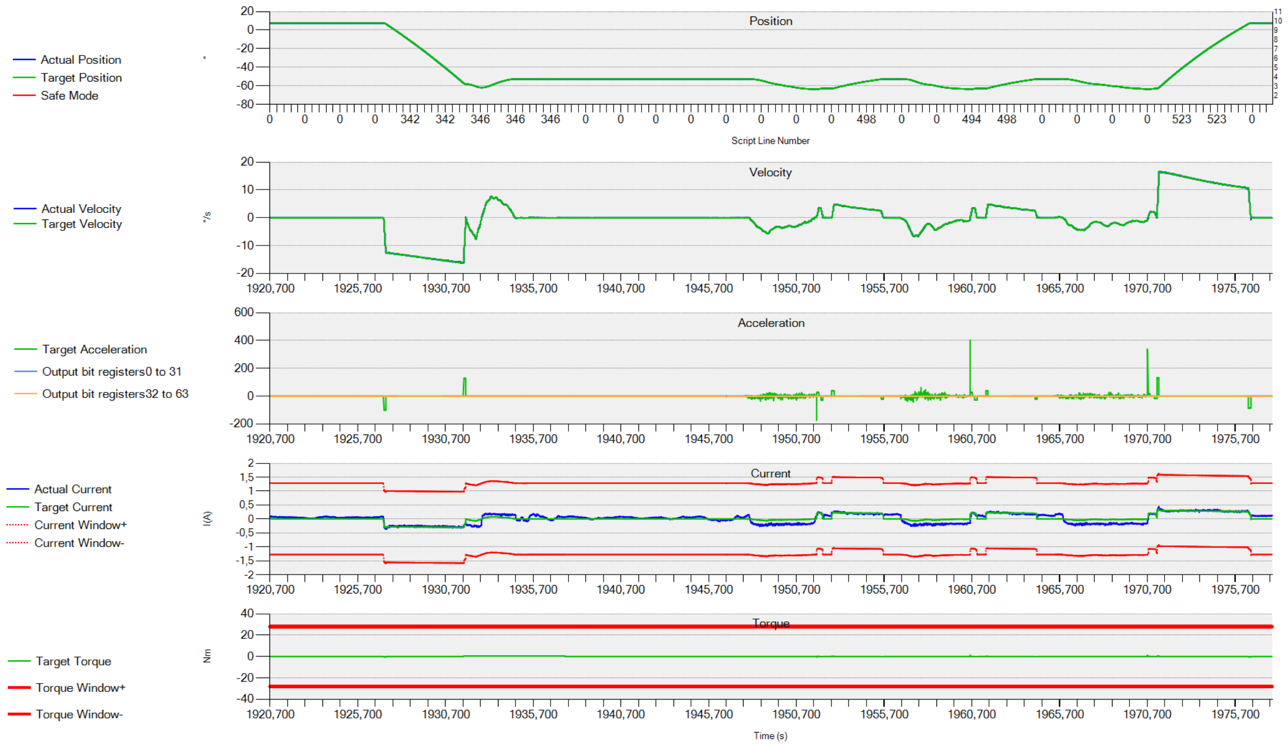


Figure 77 - Pathological subject 4. Right Side. F-Modality

Figure 78 - Pathological subject 4. Right Side. M-Modality

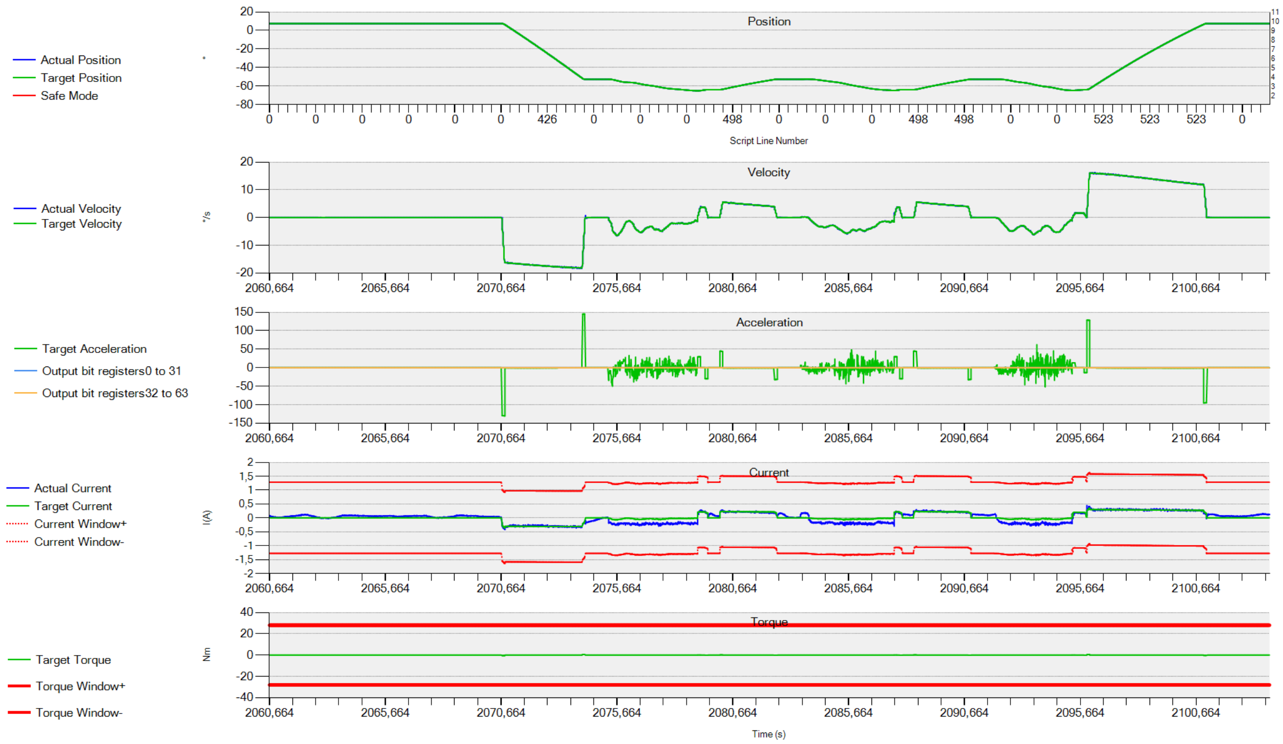


Figure 79 - Pathological subject 4. Right Side. D-Modality

PATHOLOGICAL SUBJECT 5

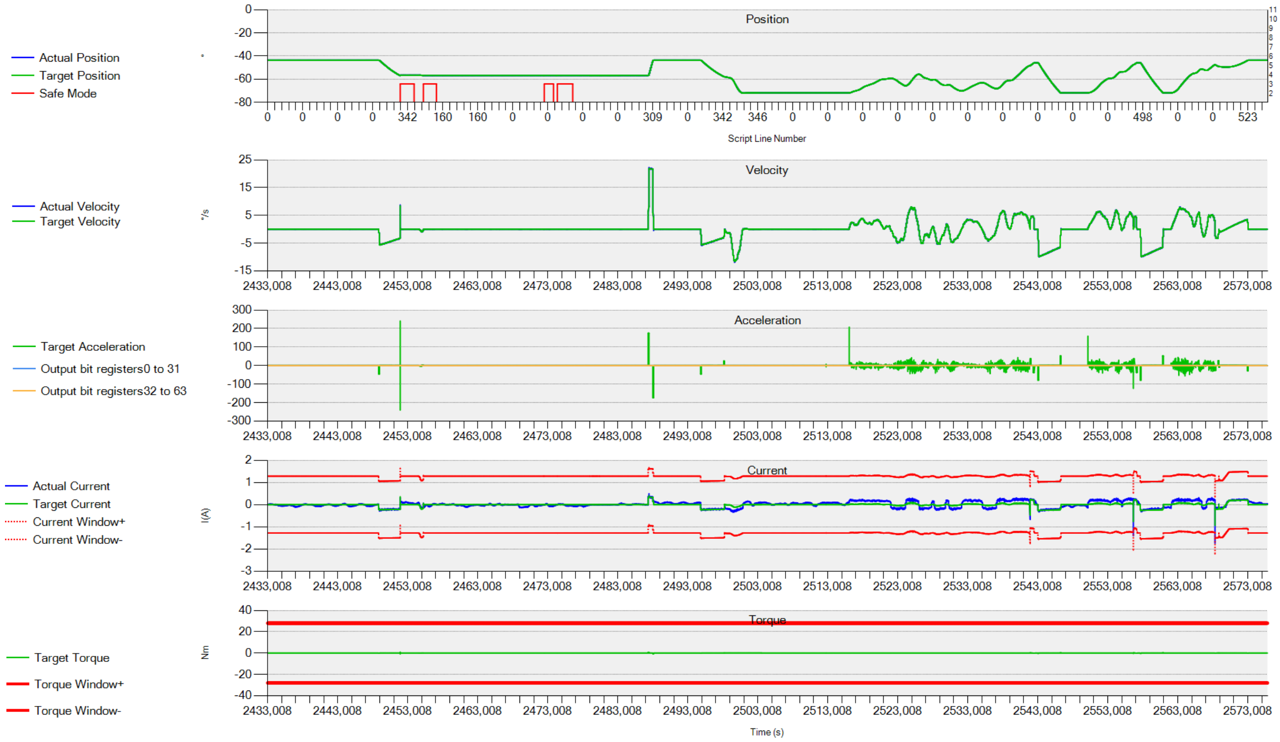


Figure 80 - Pathological subject 5. Left Side. F-Modality

Figure 81 - Pathological subject 5. Left Side. M-Modality

Figure 82 - Pathological subject 5. Left Side. D-Modality

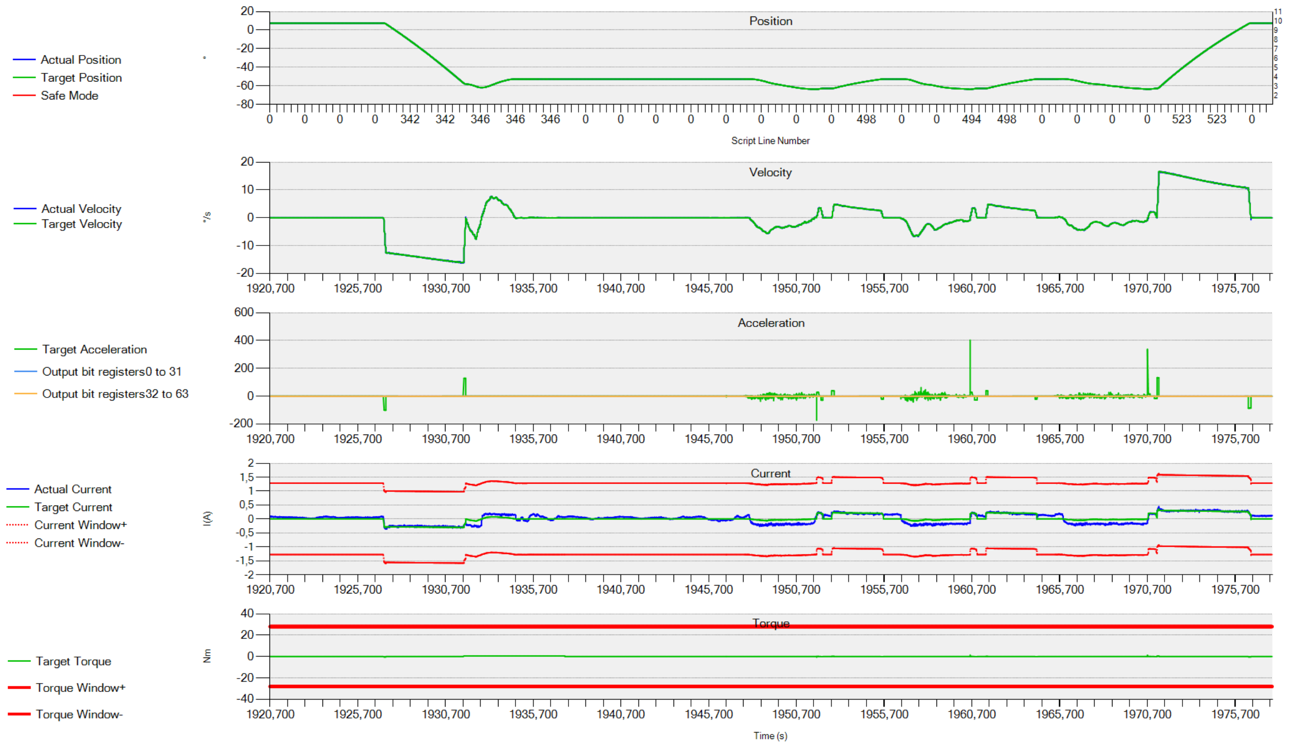


Figure 83 - Pathological subject 5. Right Side. F-Modality

Figure 84 - Pathological subject 5. Right Side. M-Modality

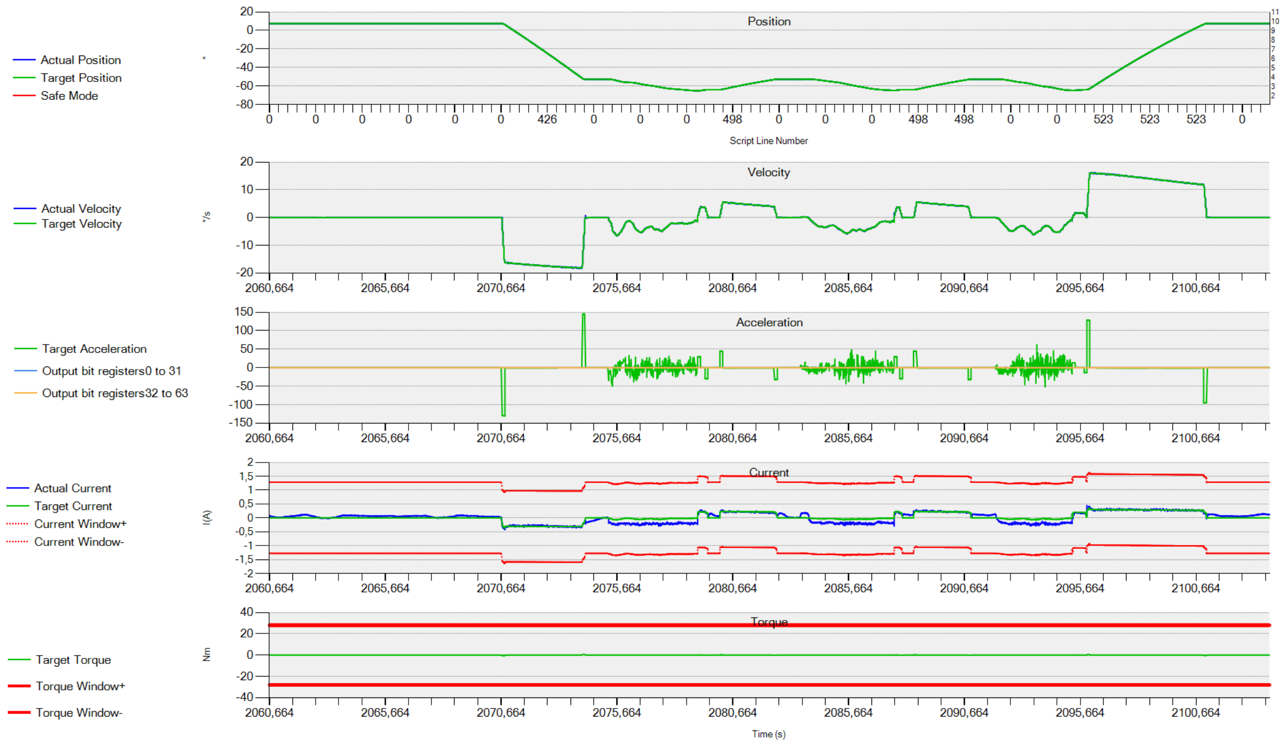


Figure 85 - Pathological subject 5. Right Side. D-Modality

PATHOLOGICAL SUBJECT 6

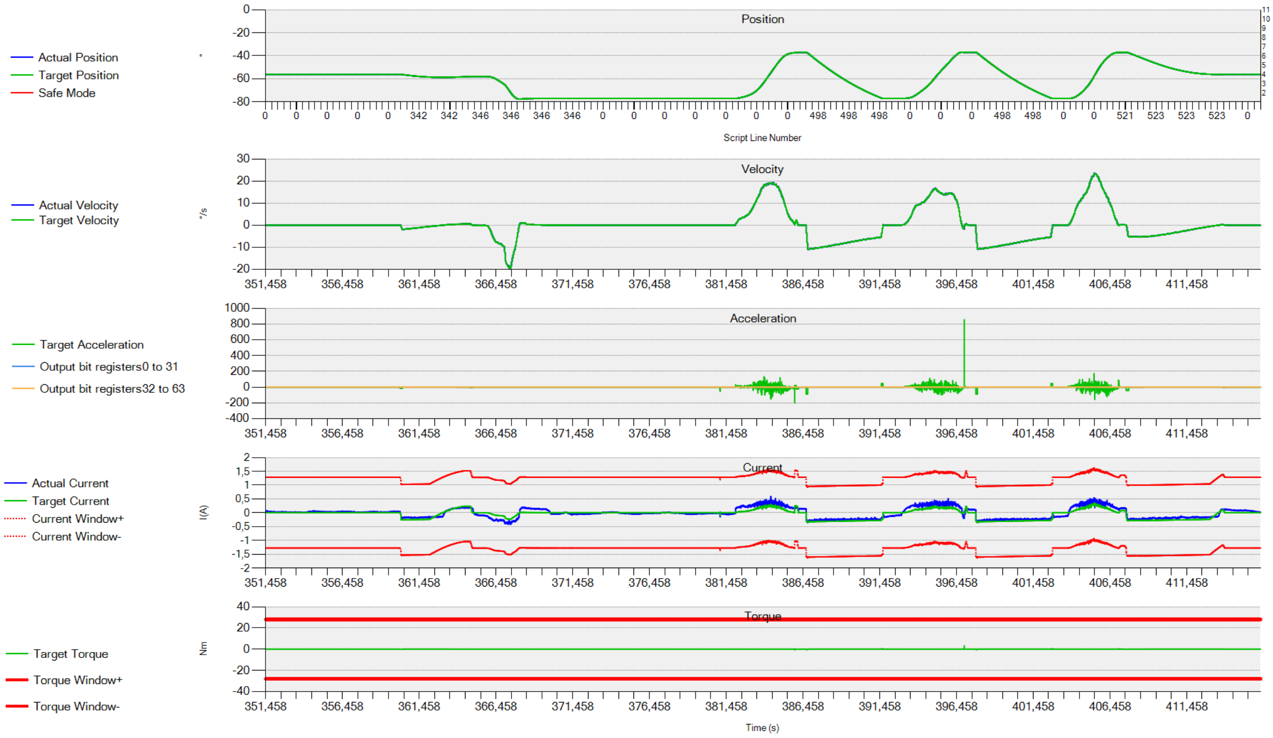


Figure 86 - Pathological subject 6. Left Side. D-Modality

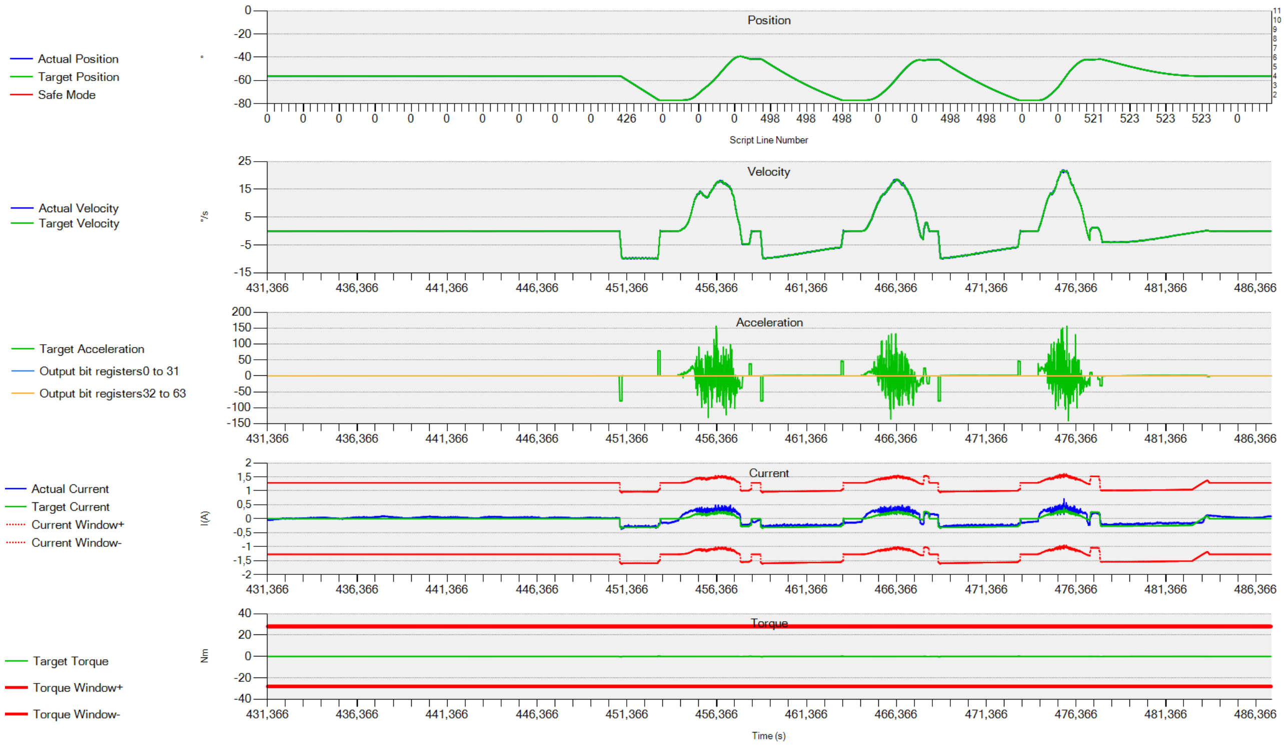


Figure 87 - Pathological subject 6. Left Side. D-Modality

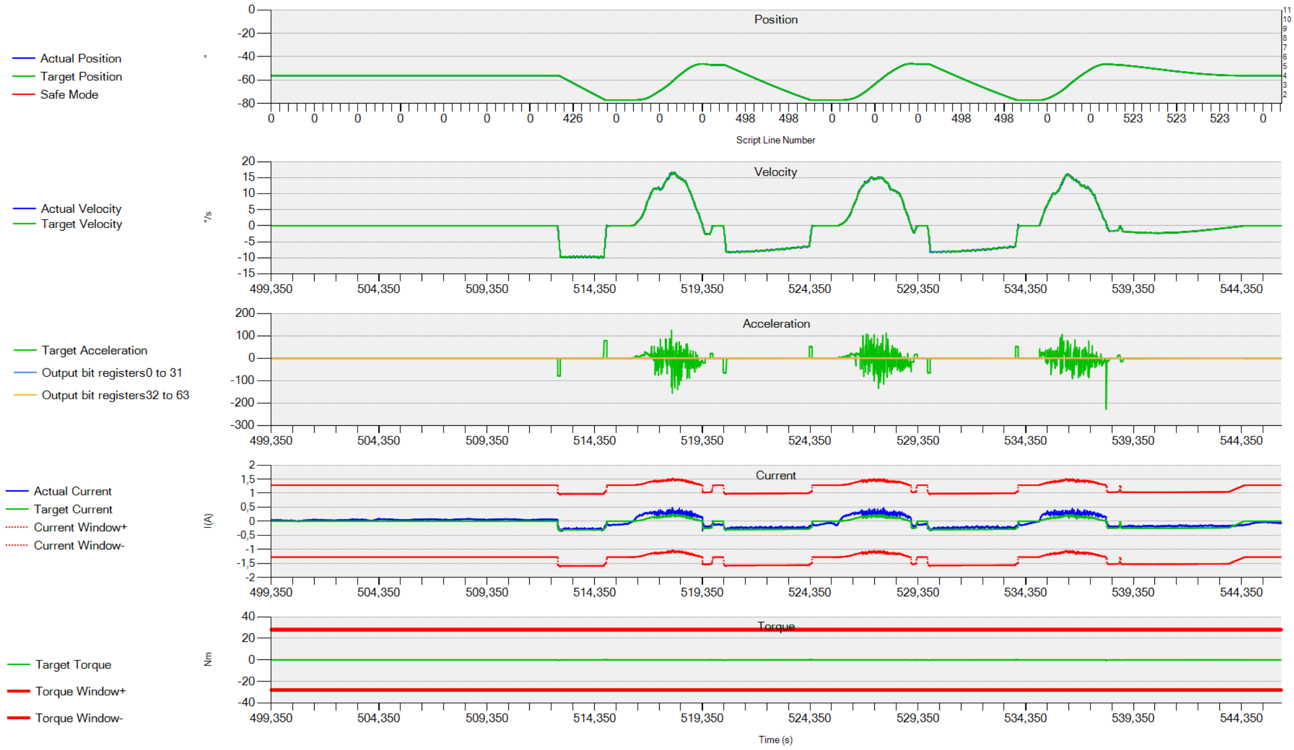


Figure 88 - Pathological subject 6. Left Side. D-Modality

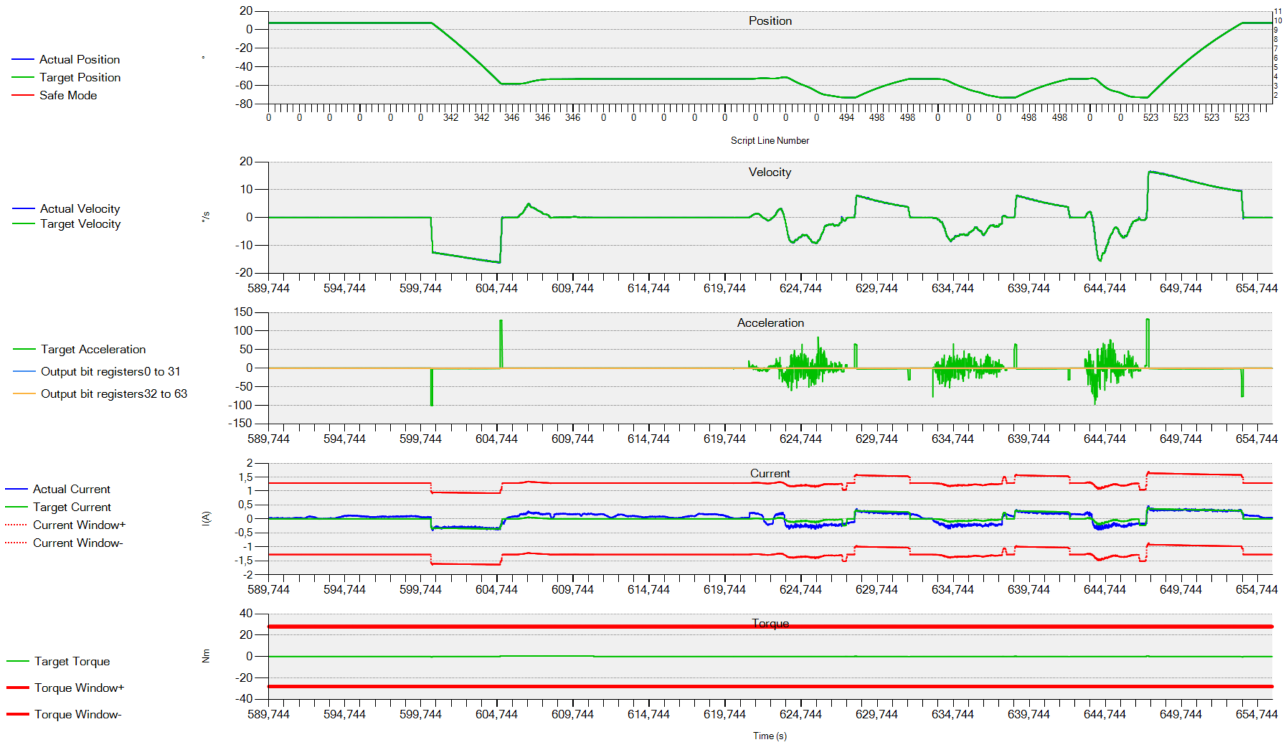


Figure 89 - Pathological subject 6. Right Side. F-Modality

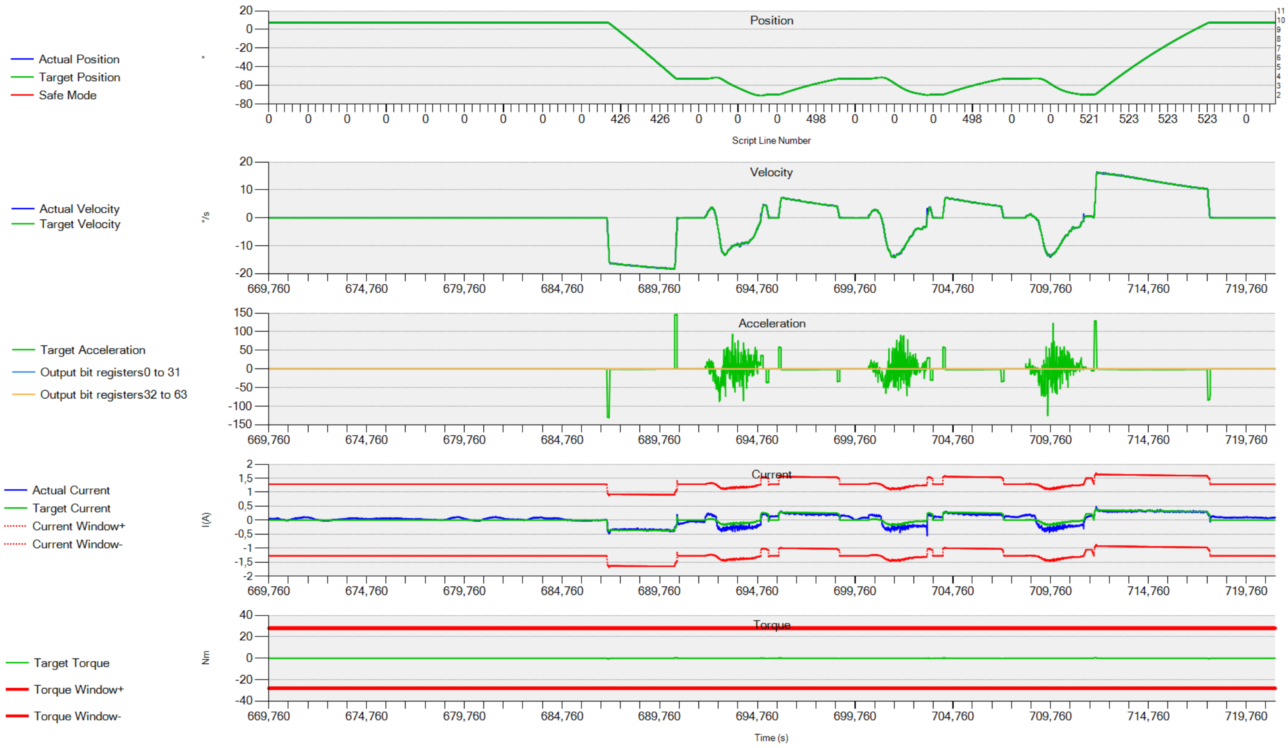


Figure 90 - Pathological subject 6. Right Side. M-Modality

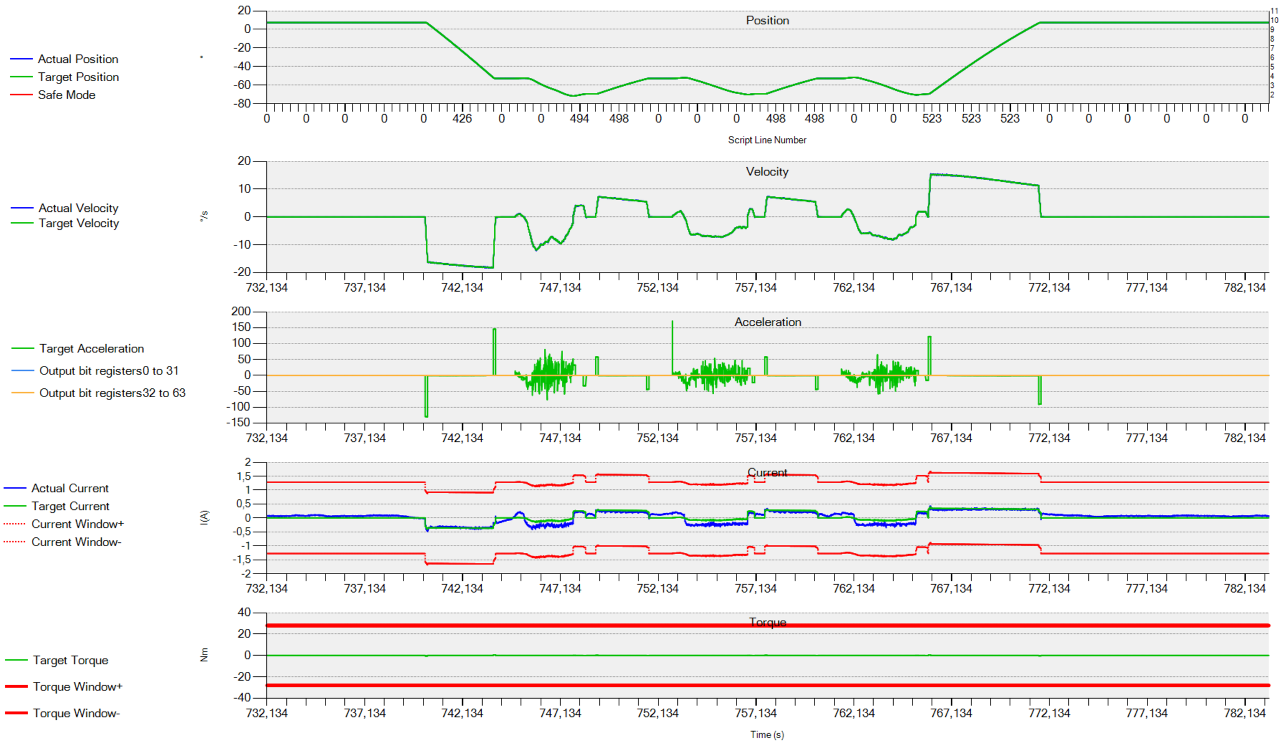


Figure 91 - Pathological subject 6. Right Side. D-Modality

PATHOLOGICAL SUBJECT 7

Figure 92 - Pathological subject 7. Left Side. F-Modality

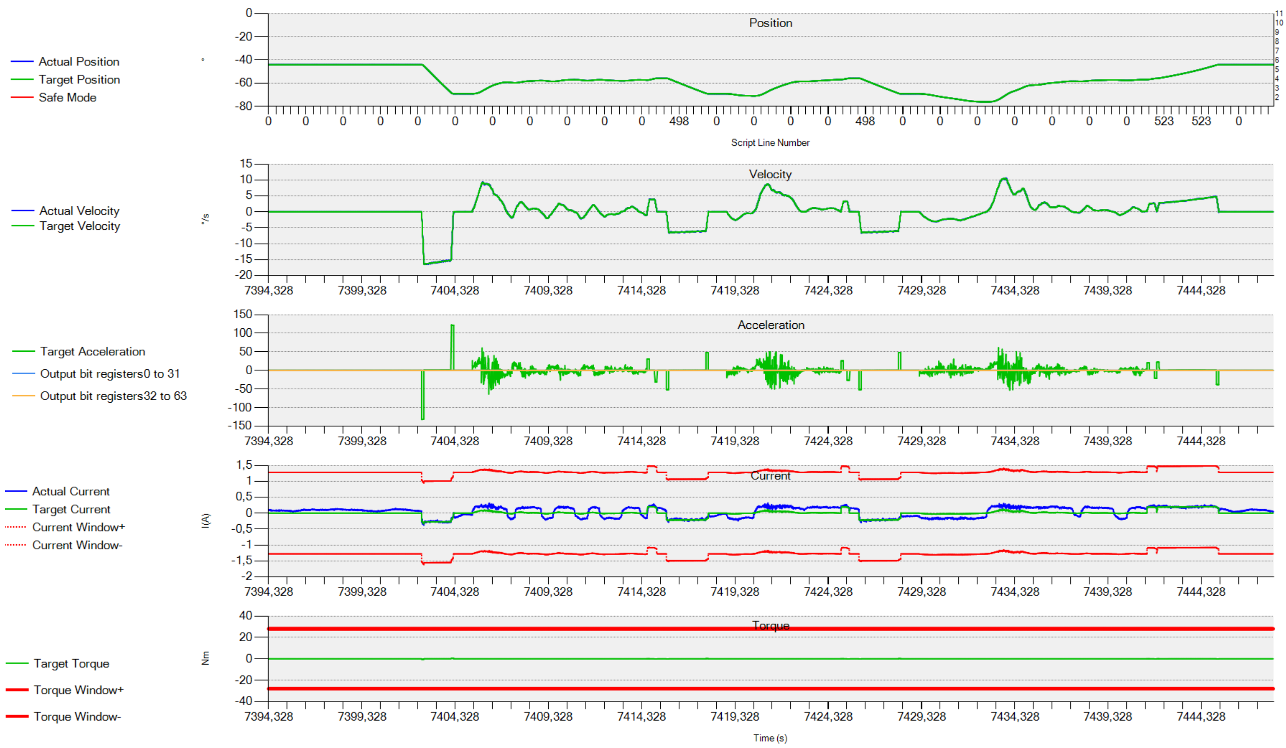


Figure 93 - Pathological subject 7. Left Side. M-Modality

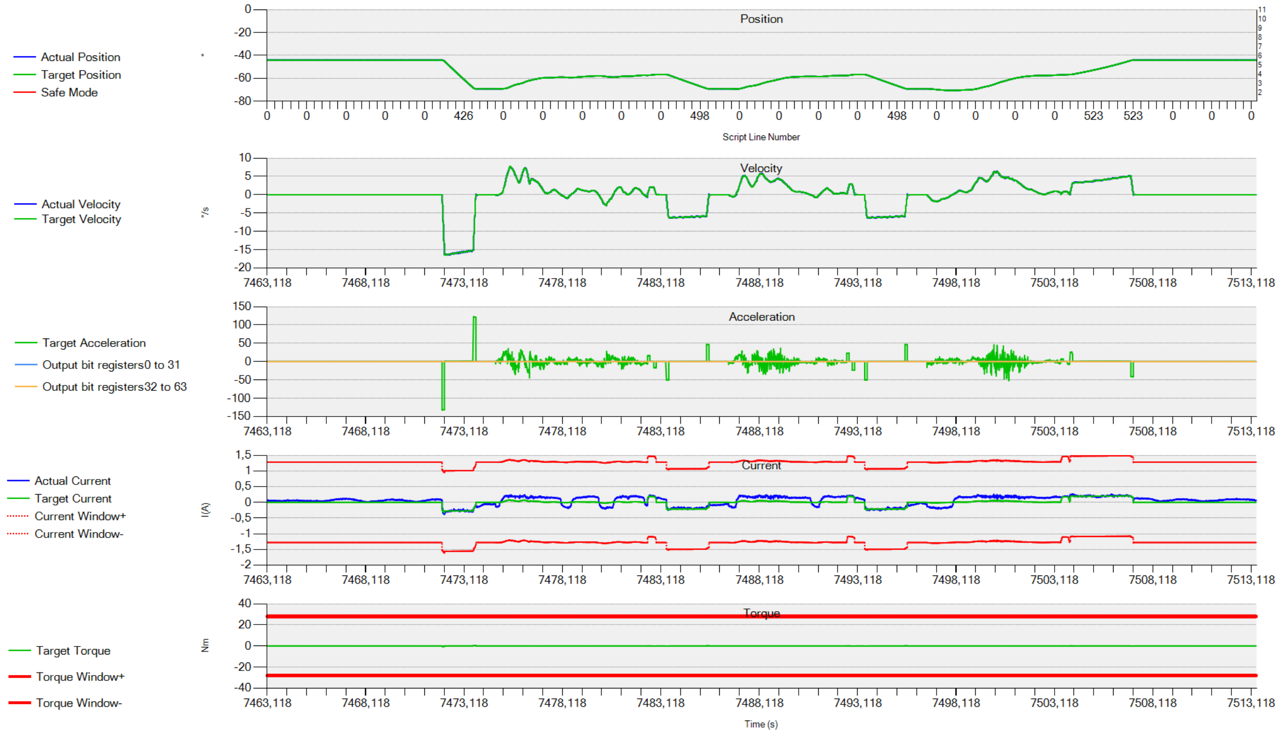


Figure 94 - Pathological subject 7. Left Side. D-Modality

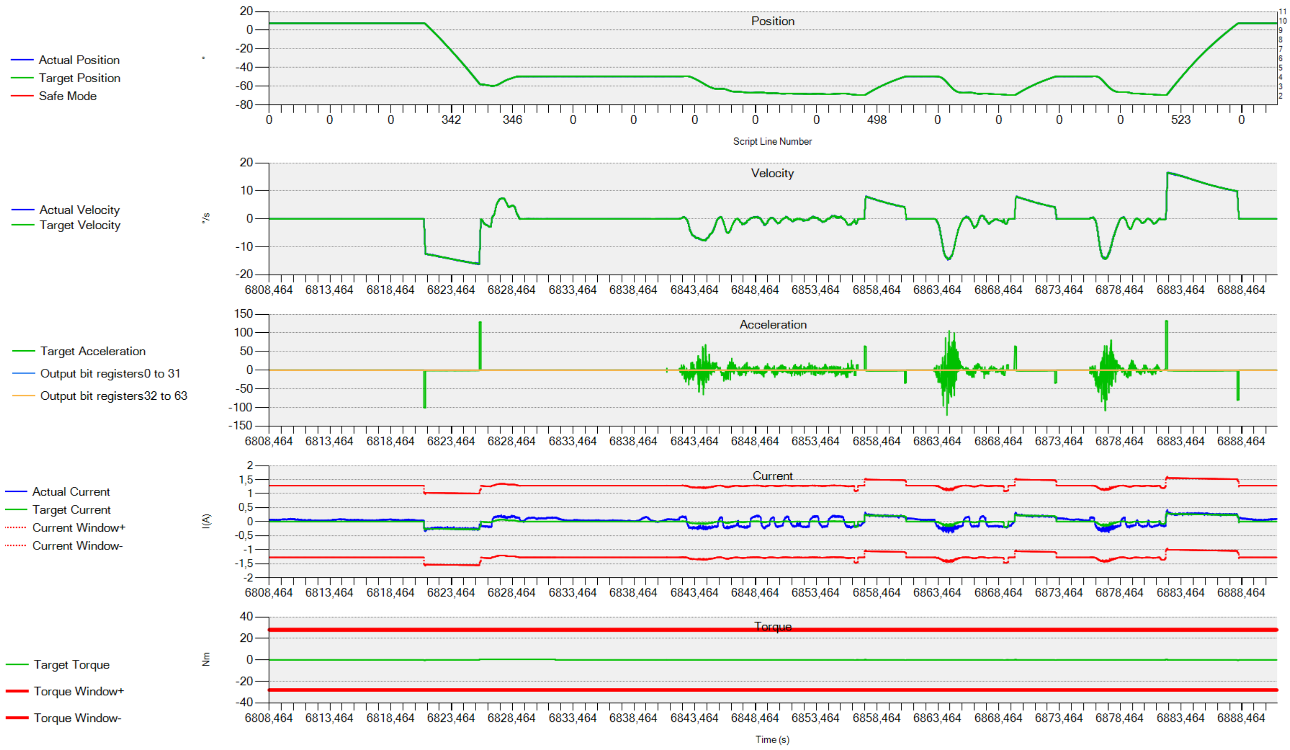


Figure 95 - Pathological subject 7. Right Side. F-Modality

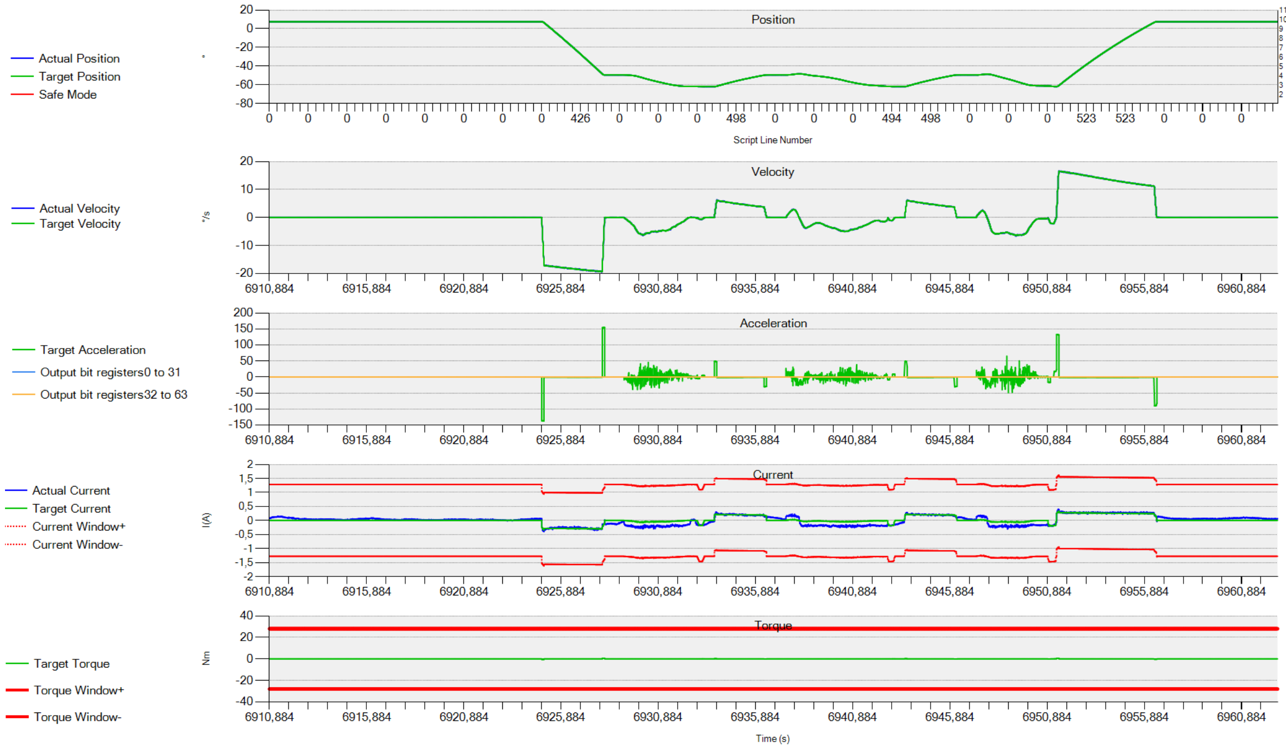


Figure 96 - Pathological subject 7. Right Side. M-Modality

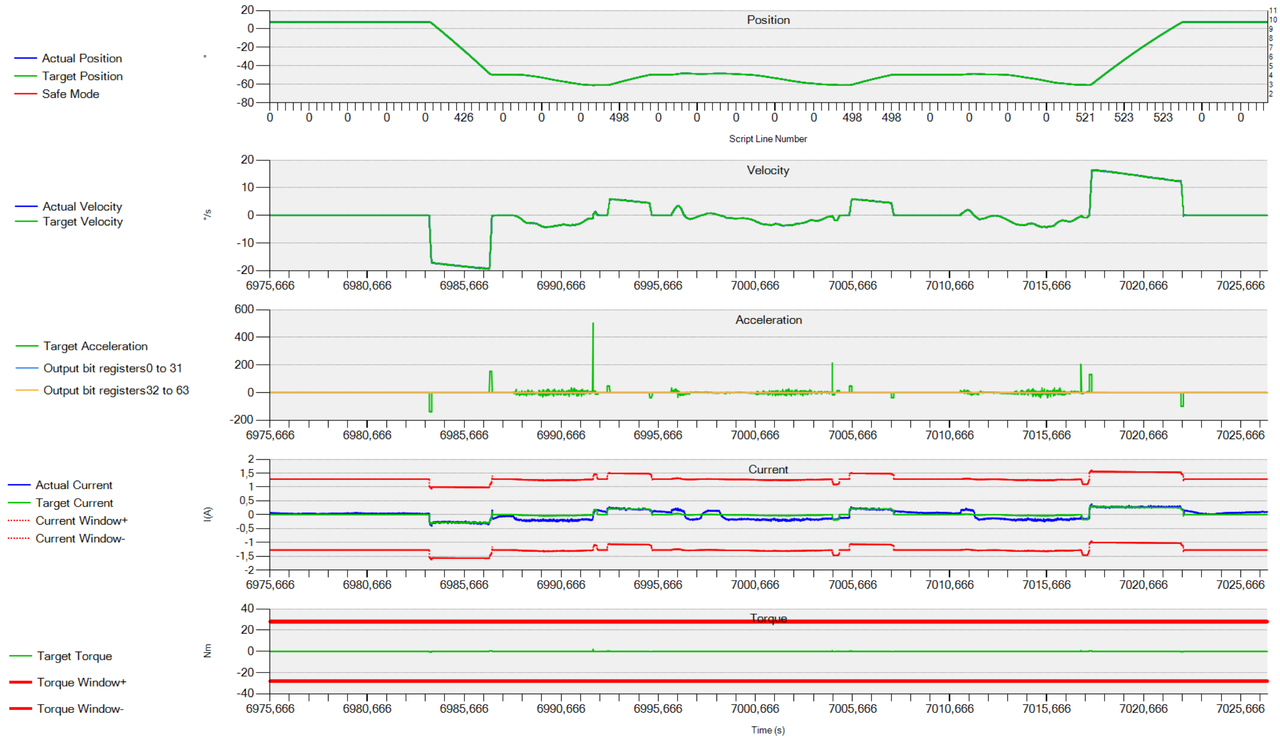


Figure 97 - Pathological subject 7. Right Side. D-Modality

PATHOLOGICAL SUBJECT 8

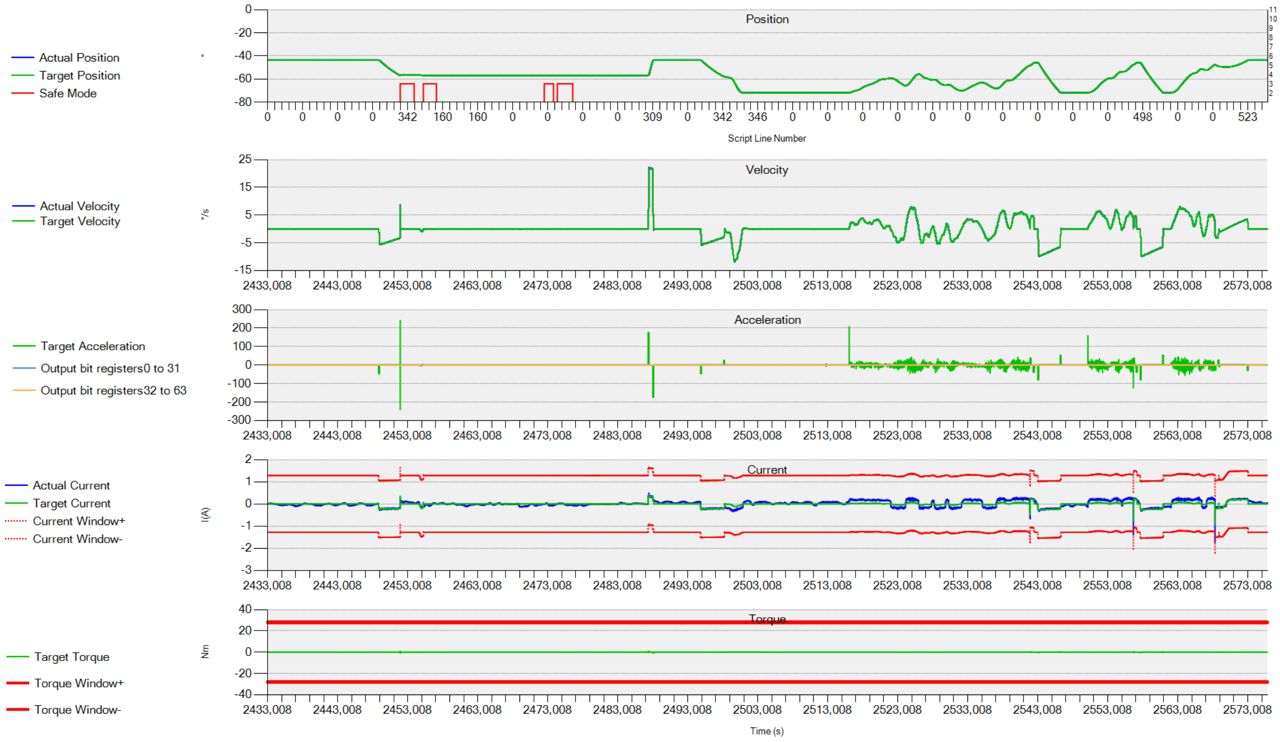


Figure 98 - Pathological subject 8. Left Side. F-Modality

Figure 99 - Pathological subject 8. Left Side. M-Modality

Figure 100 - Pathological subject 8. Left Side. D-Modality

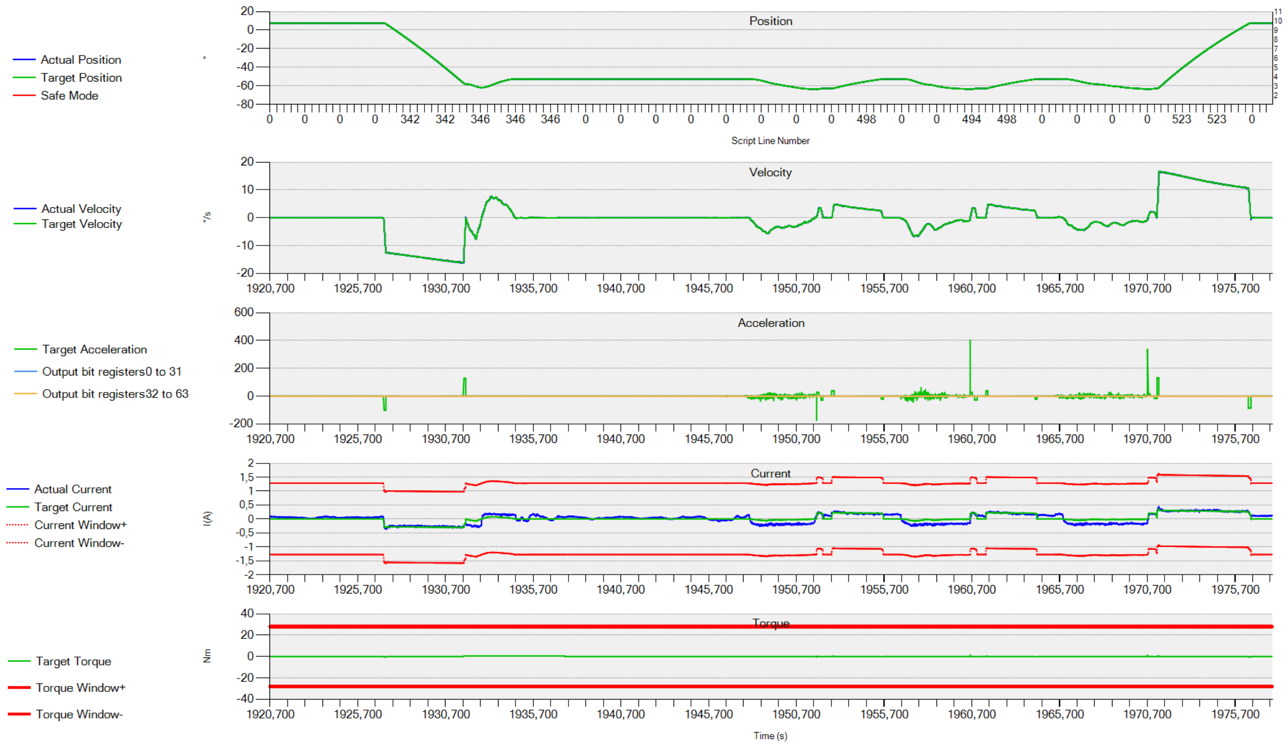


Figure 101 - Pathological subject 8. Right Side. F-Modality

Figure 102 - Pathological subject 8. Right Side. M-Modality

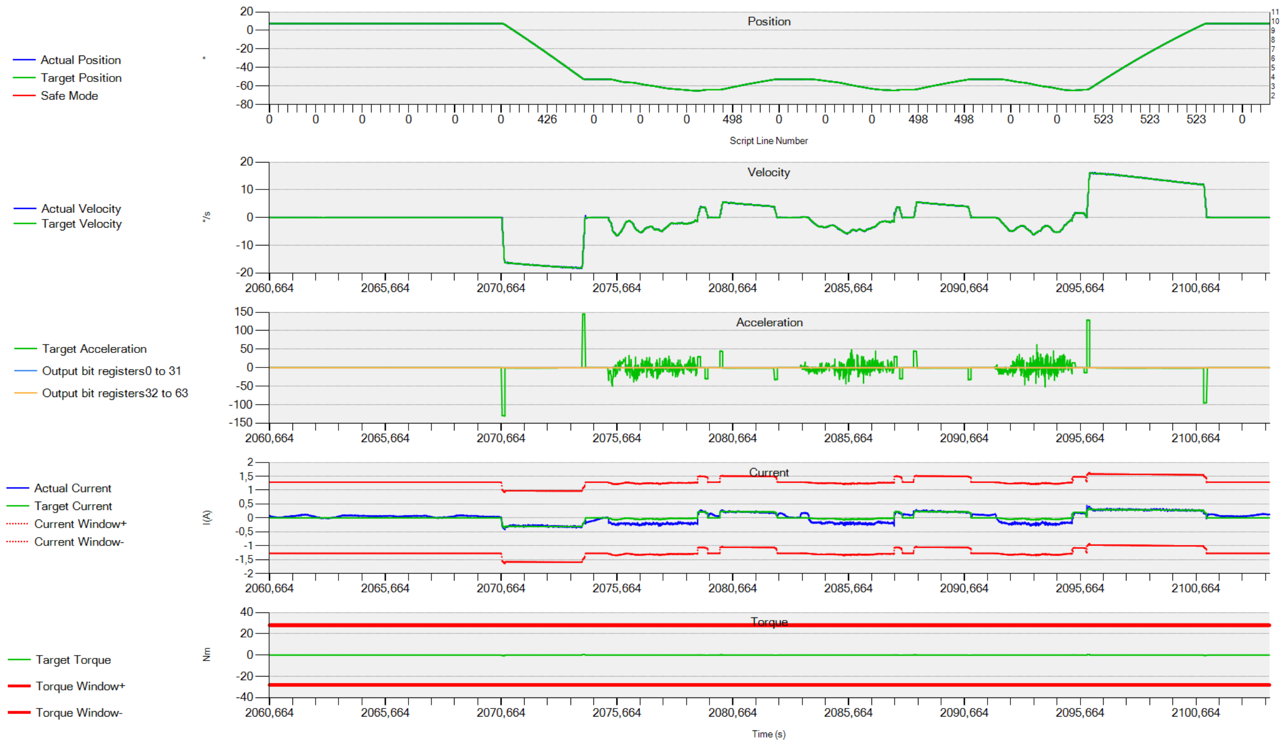


Figure 103 - Pathological subject 8. Right Side. D-Modality

PATHOLOGICAL SUBJECT 9

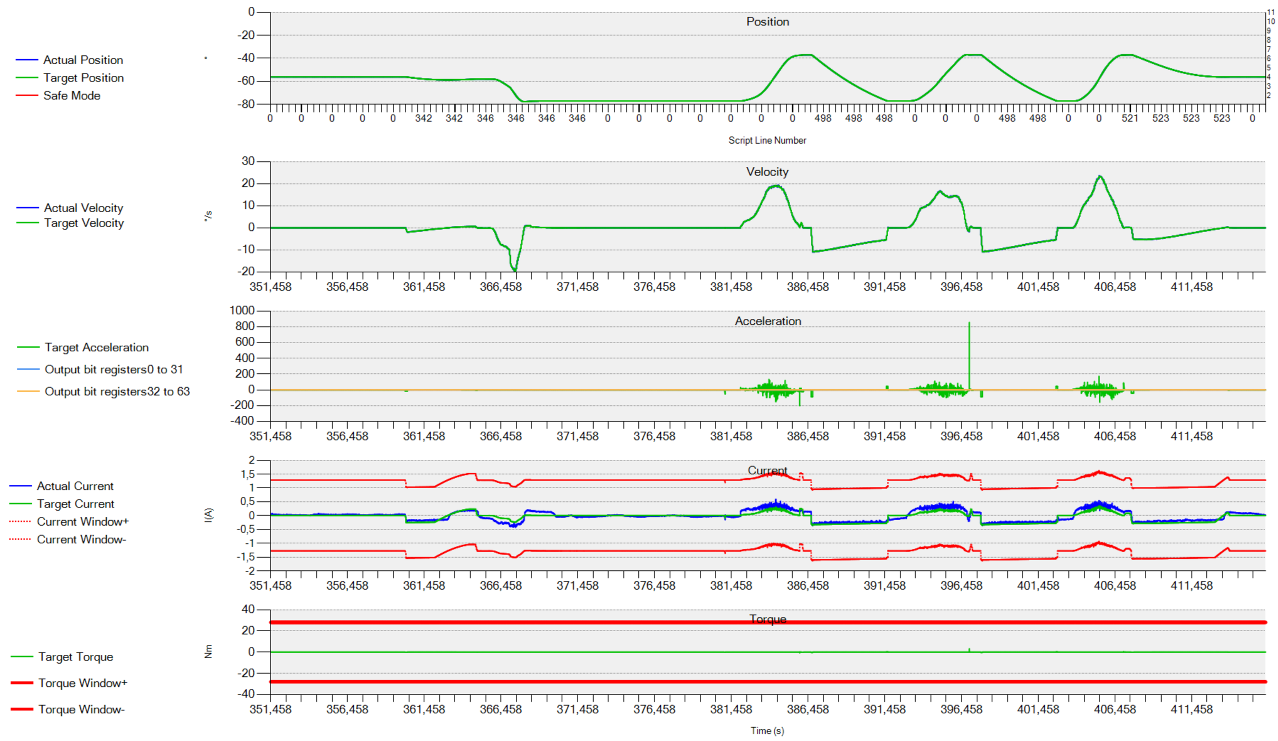


Figure 104 - Pathological subject 9. Left Side. D-Modality

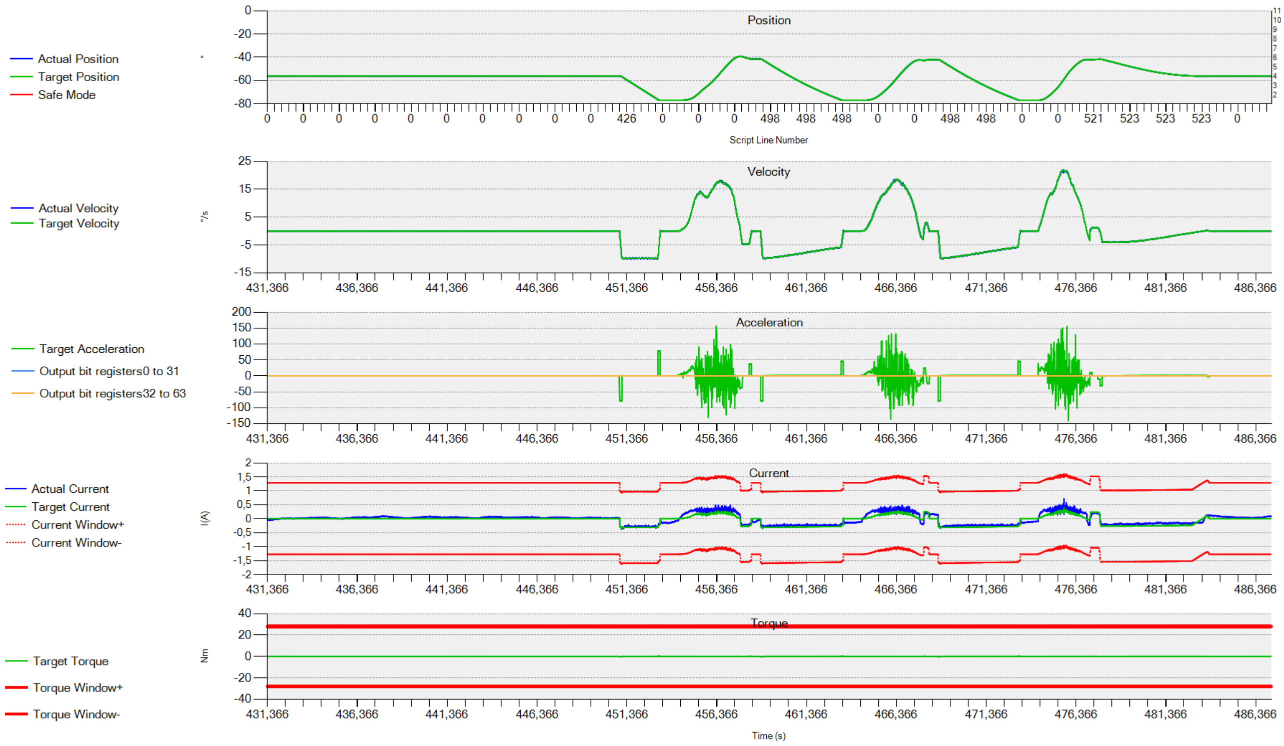


Figure 105 - Pathological subject 9. Left Side. D-Modality

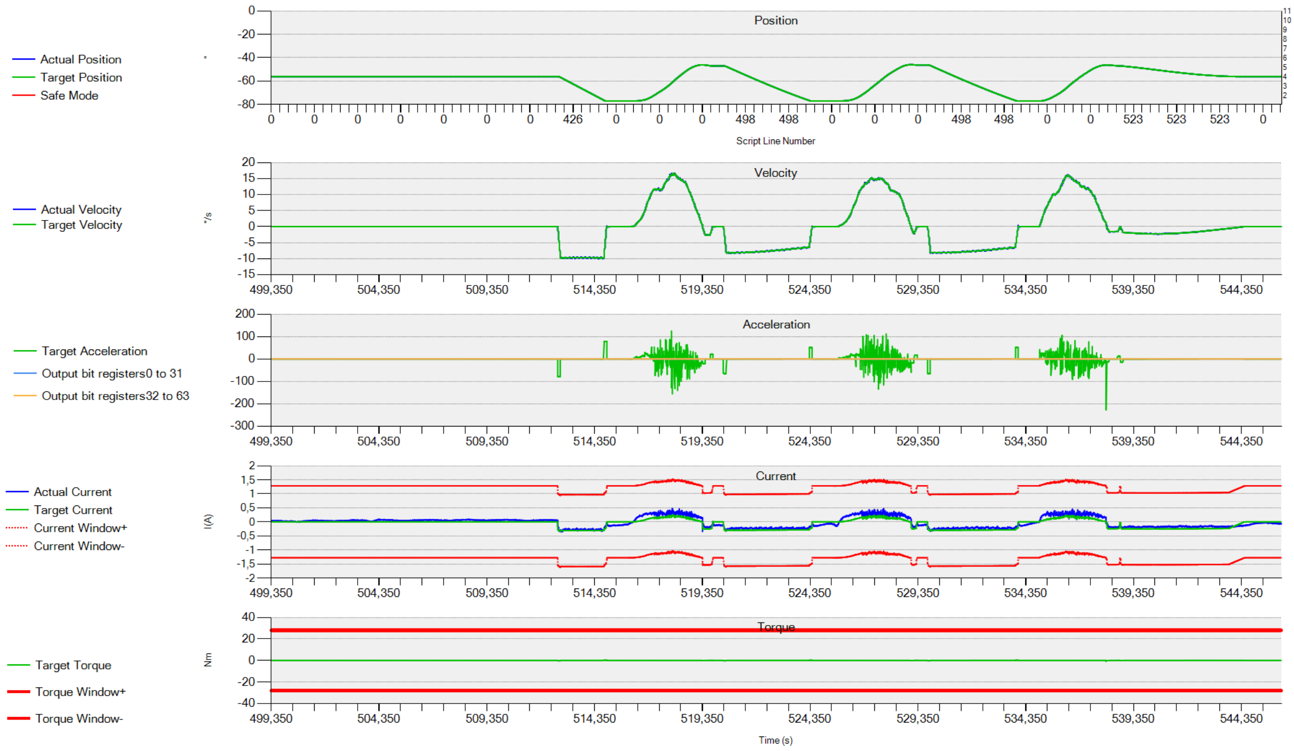


Figure 106 - Pathological subject 9. Left Side. D-Modality

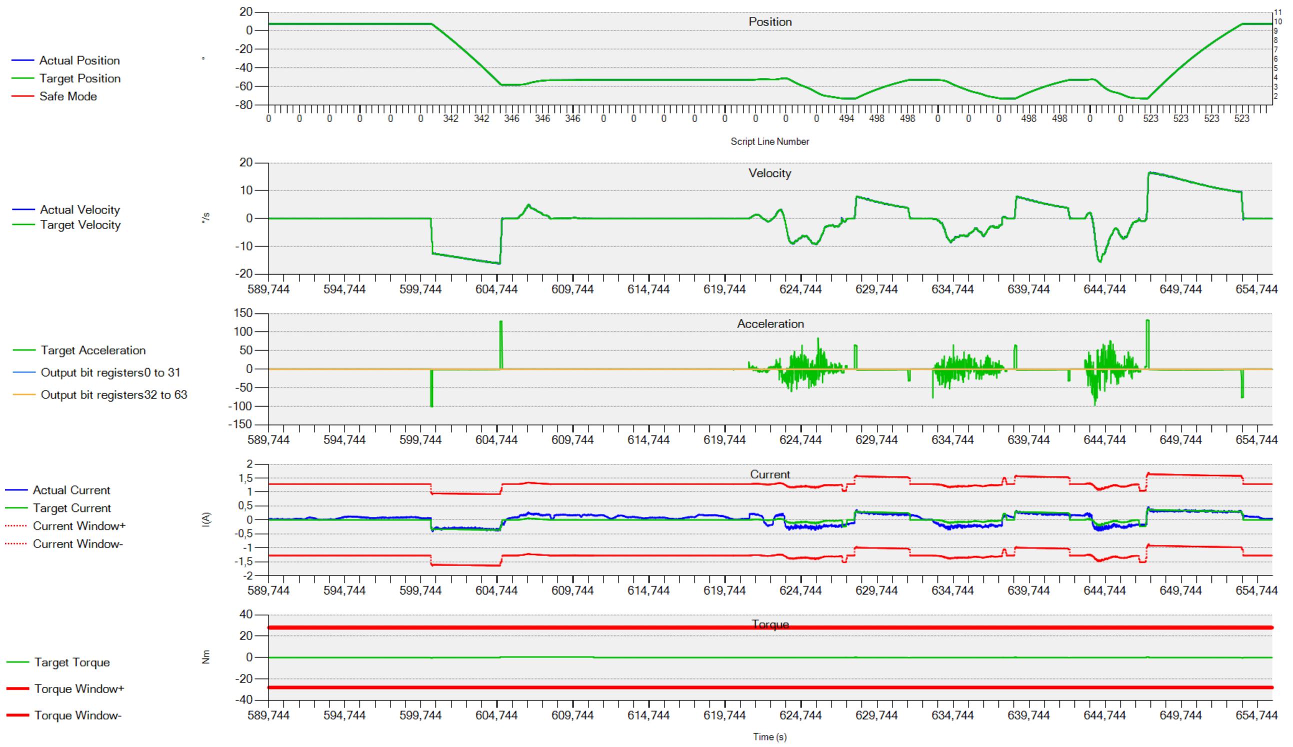


Figure 107 - Pathological subject 9. Right Side. F-Modality

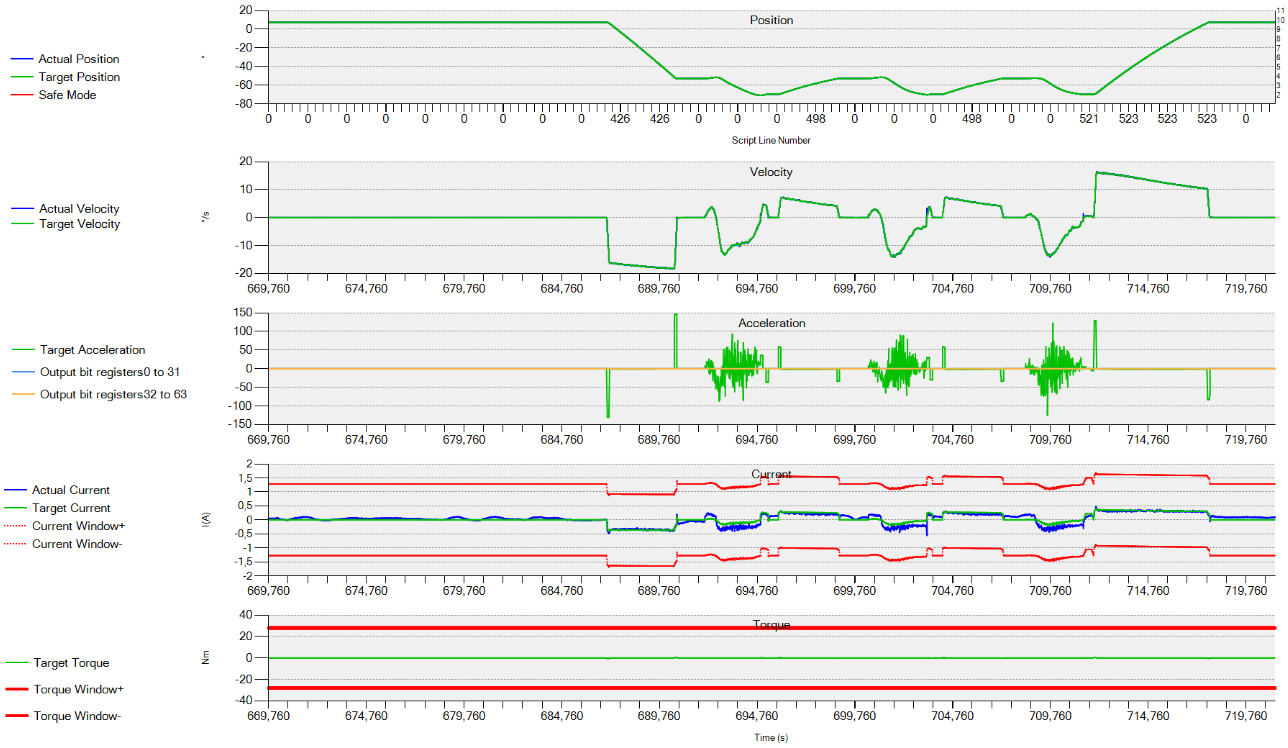


Figure 108 - Pathological subject 9. Right Side. M-Modality

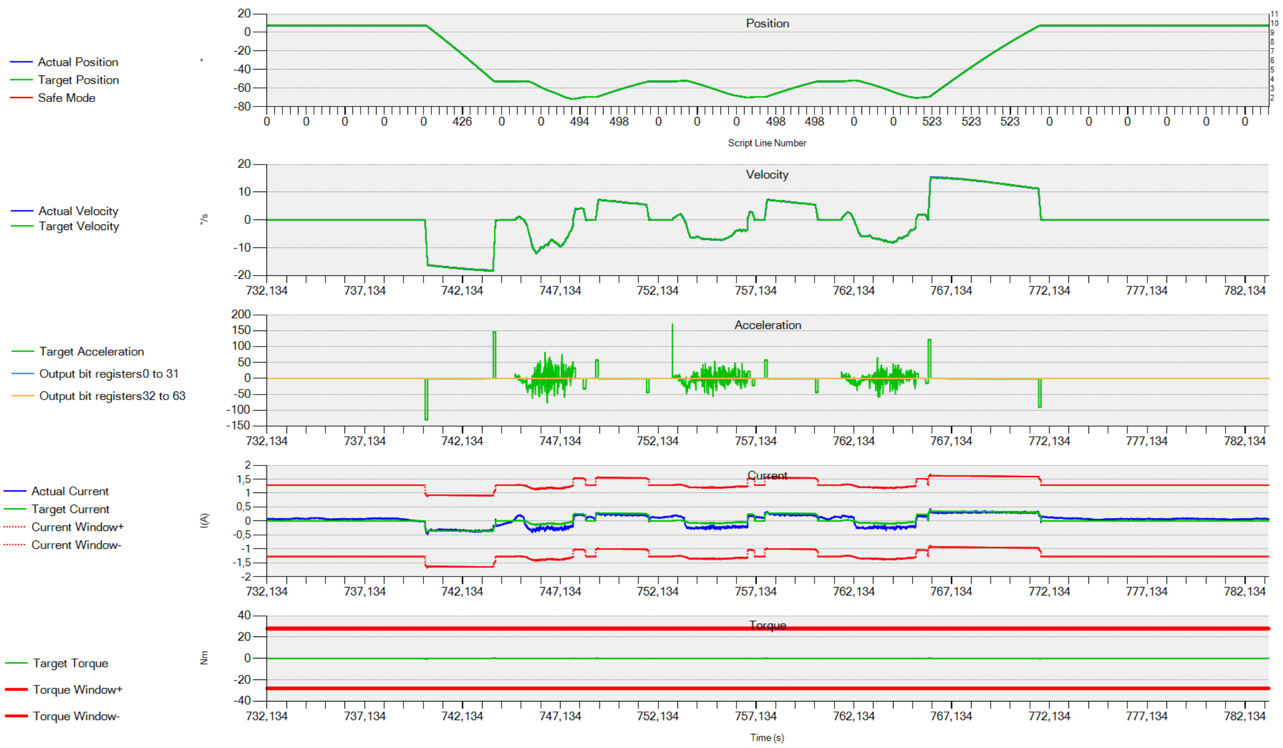


Figure 109 - Pathological subject 9. Right Side. D-Modality

PATHOLOGICAL SUBJECT 10

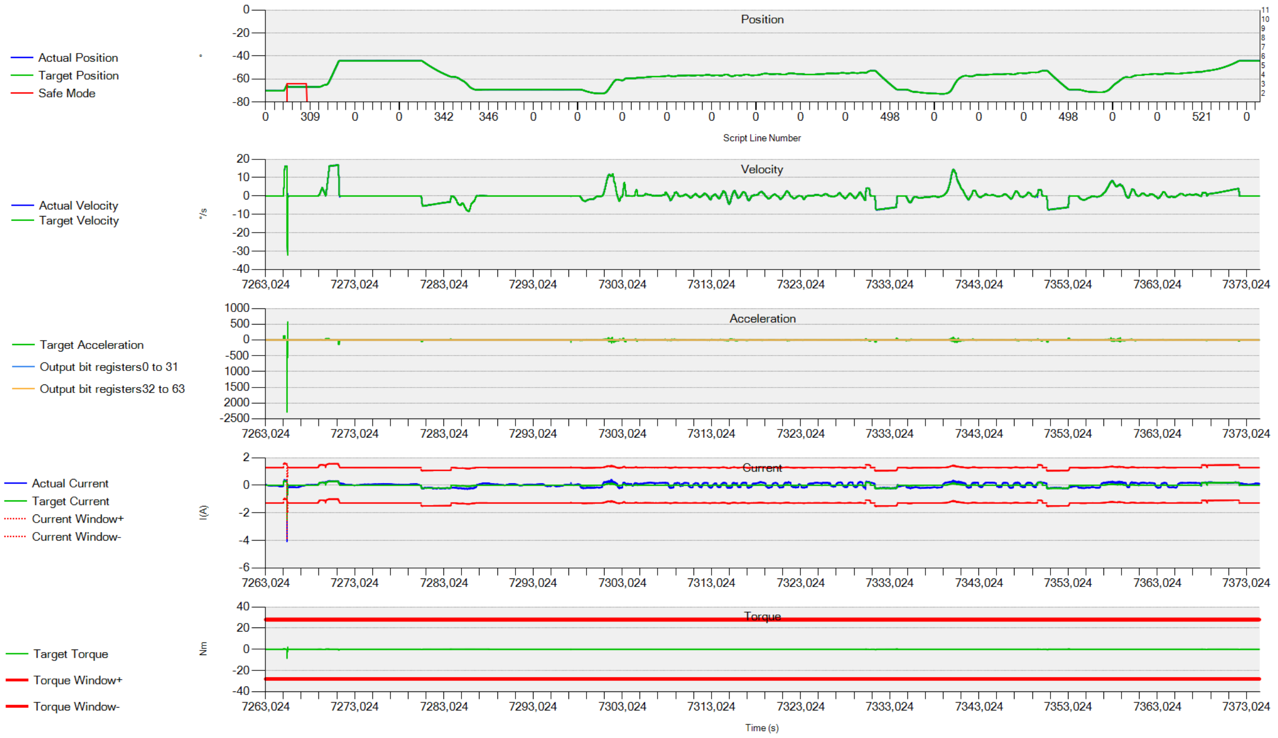


Figure 110 - Pathological subject 10. Left Side. F-Modality

Figure 111 - Pathological subject 10. Left Side. M-Modality

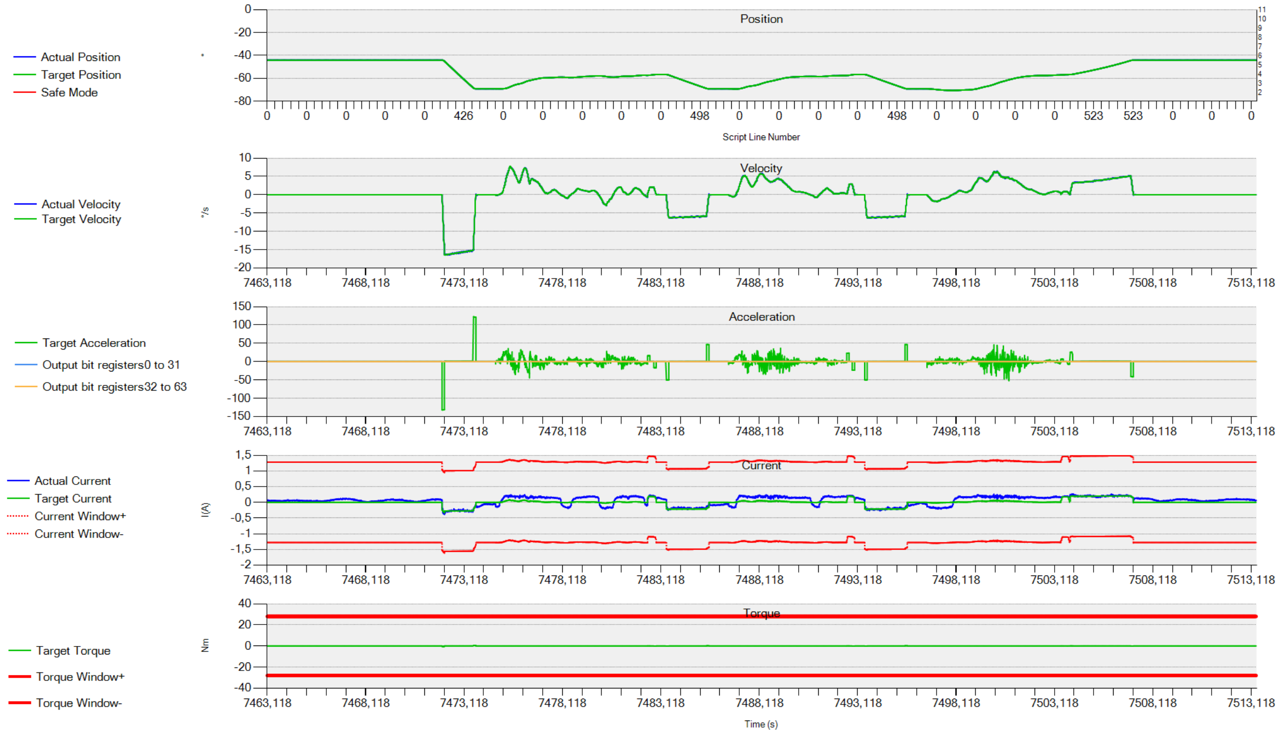


Figure 112 - Pathological subject 10. Left Side. D-Modality

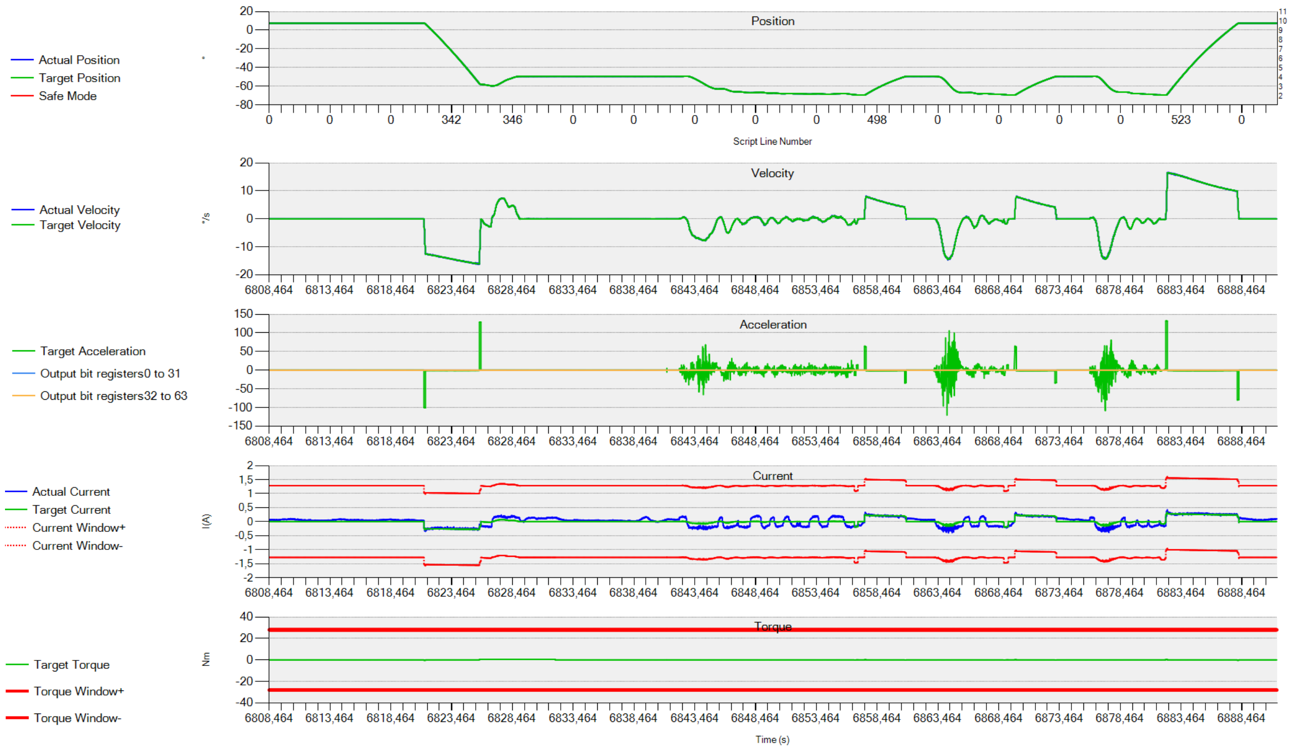


Figure 113 - Pathological subject 10. Right Side. F-Modality

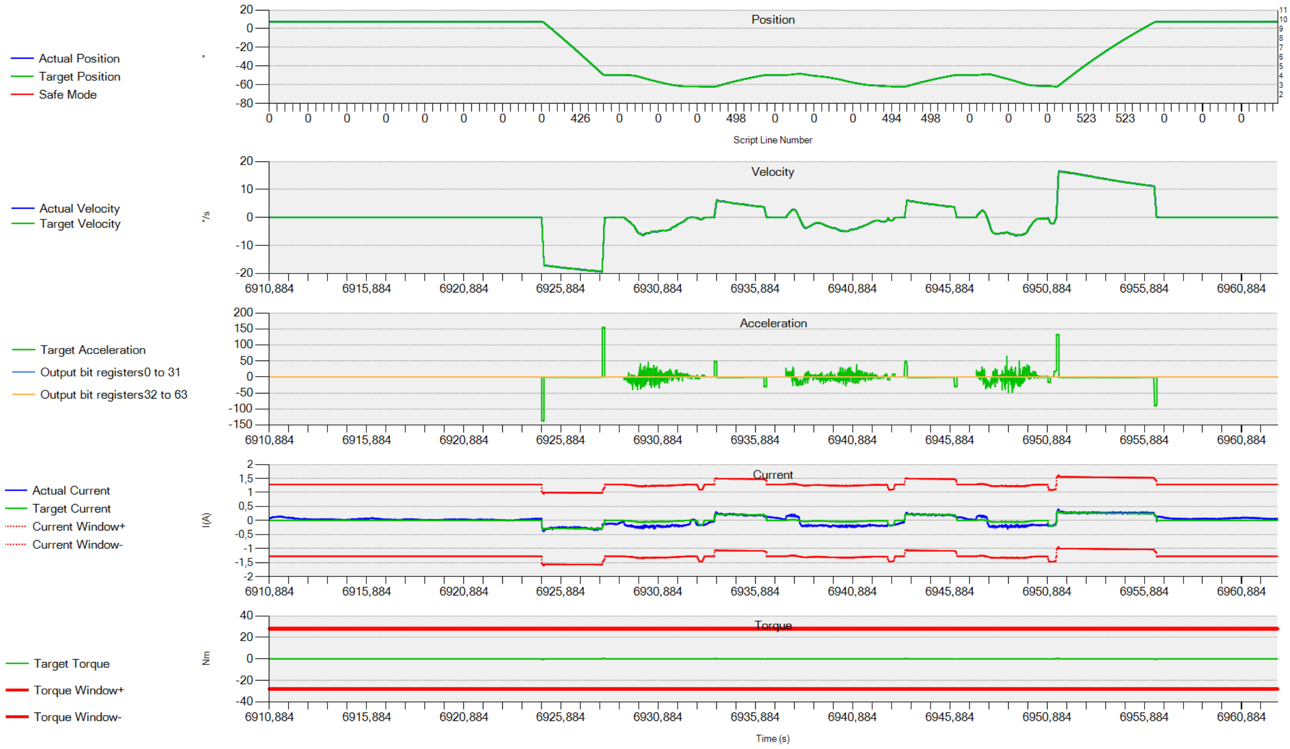


Figure 114 - Pathological subject 10. Right Side. M-Modality

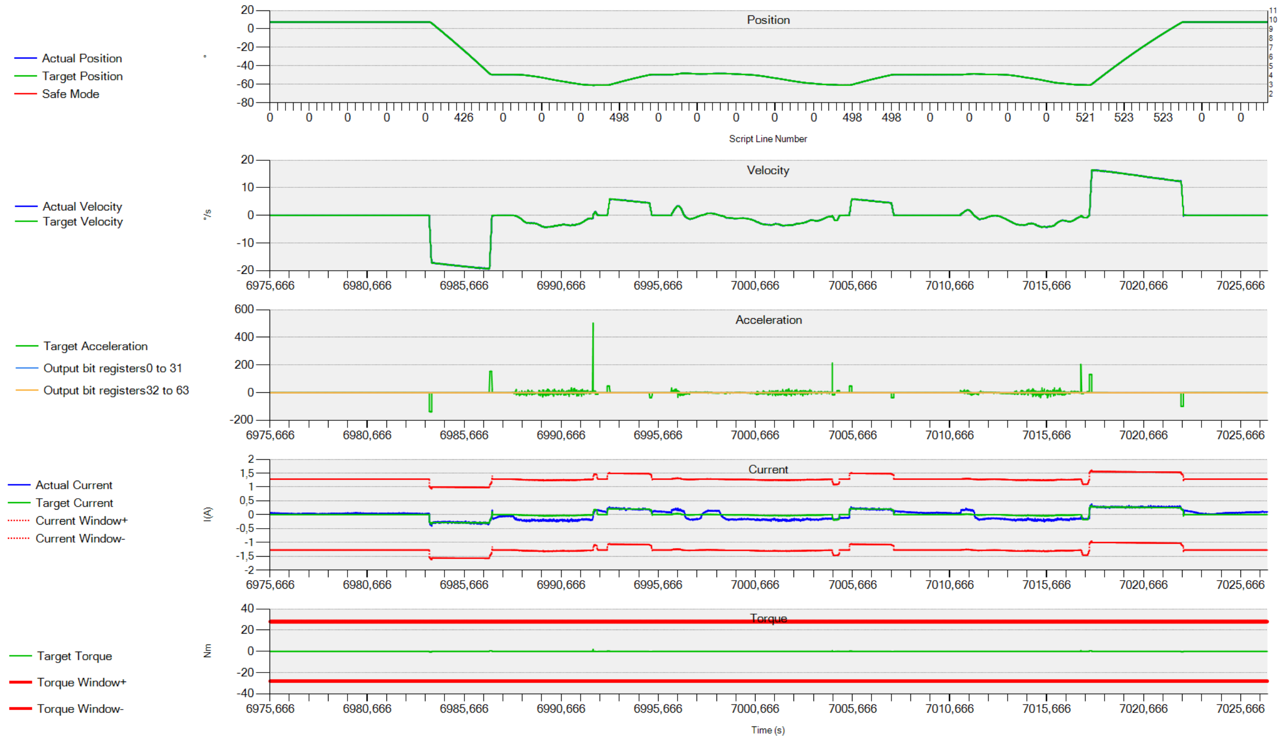


Figure 115 - Pathological subject 10. Right Side. D-Modality

HEALTHY SUBJECT 12

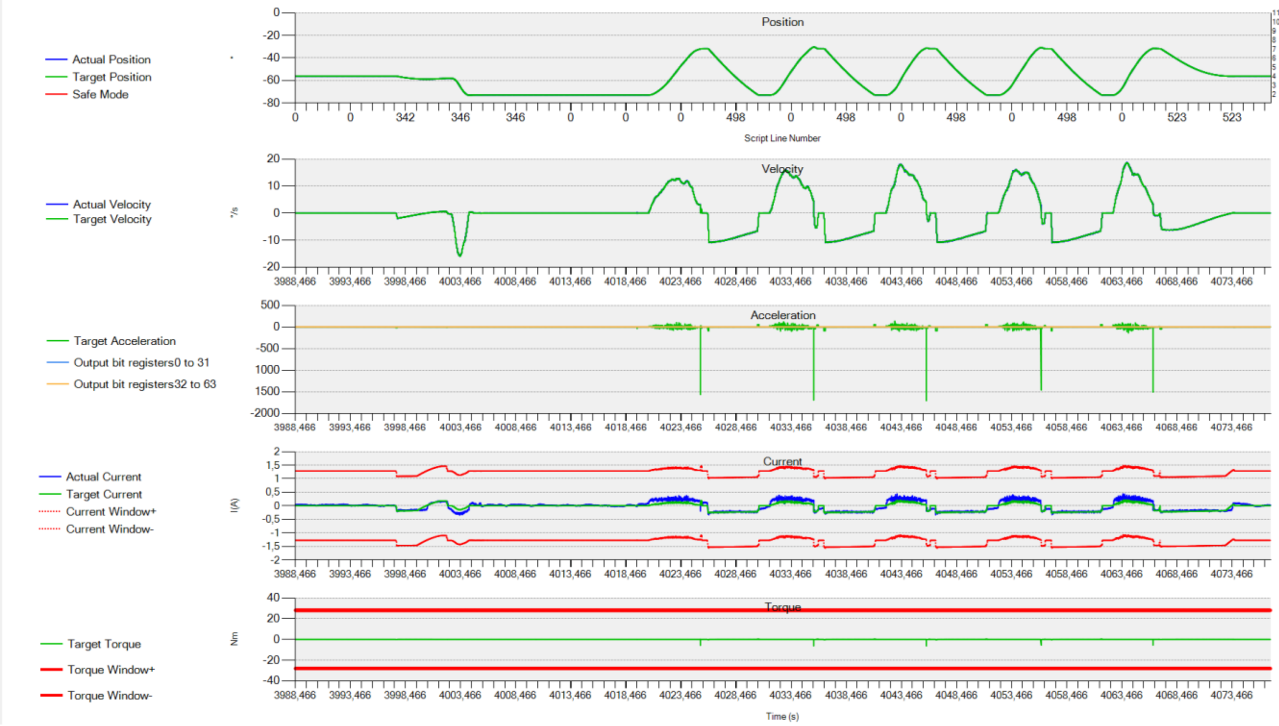


Figure 116 - Healthy subject 12. Left Side. F-Modality

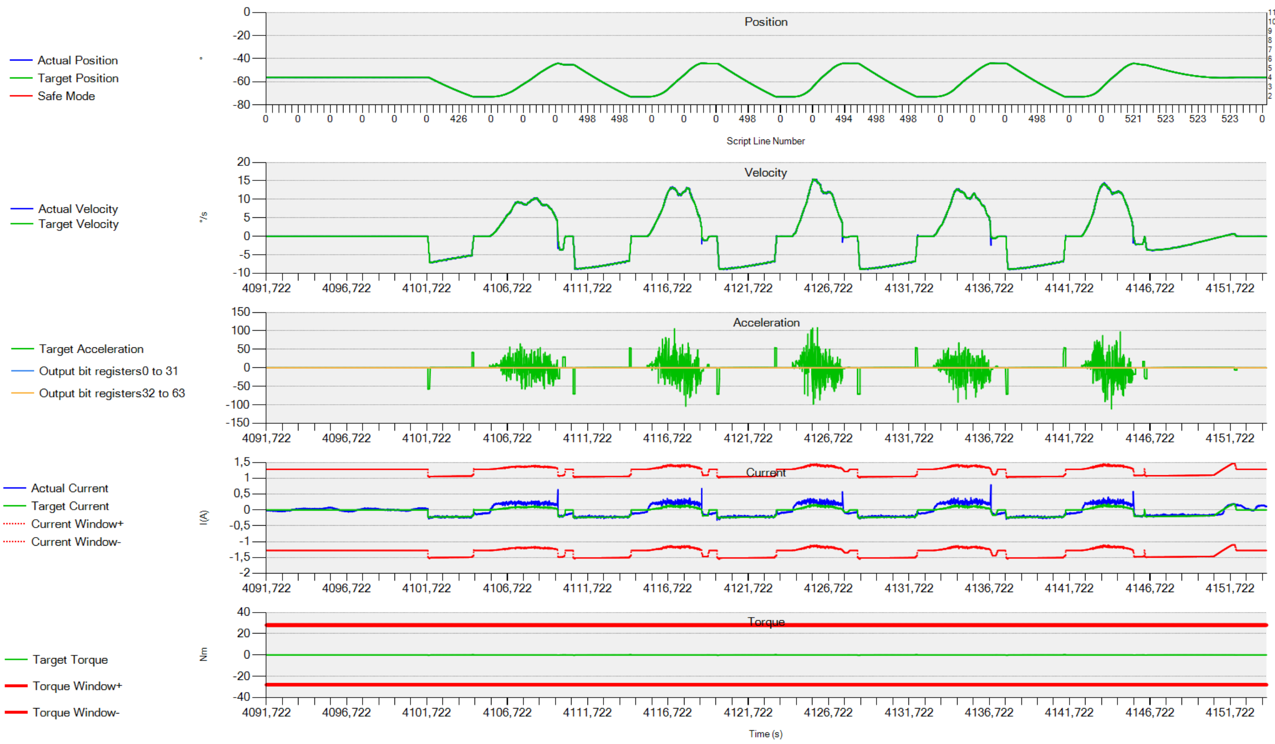


Figure 117 - Healthy subject 12. Left Side. M-Modality.

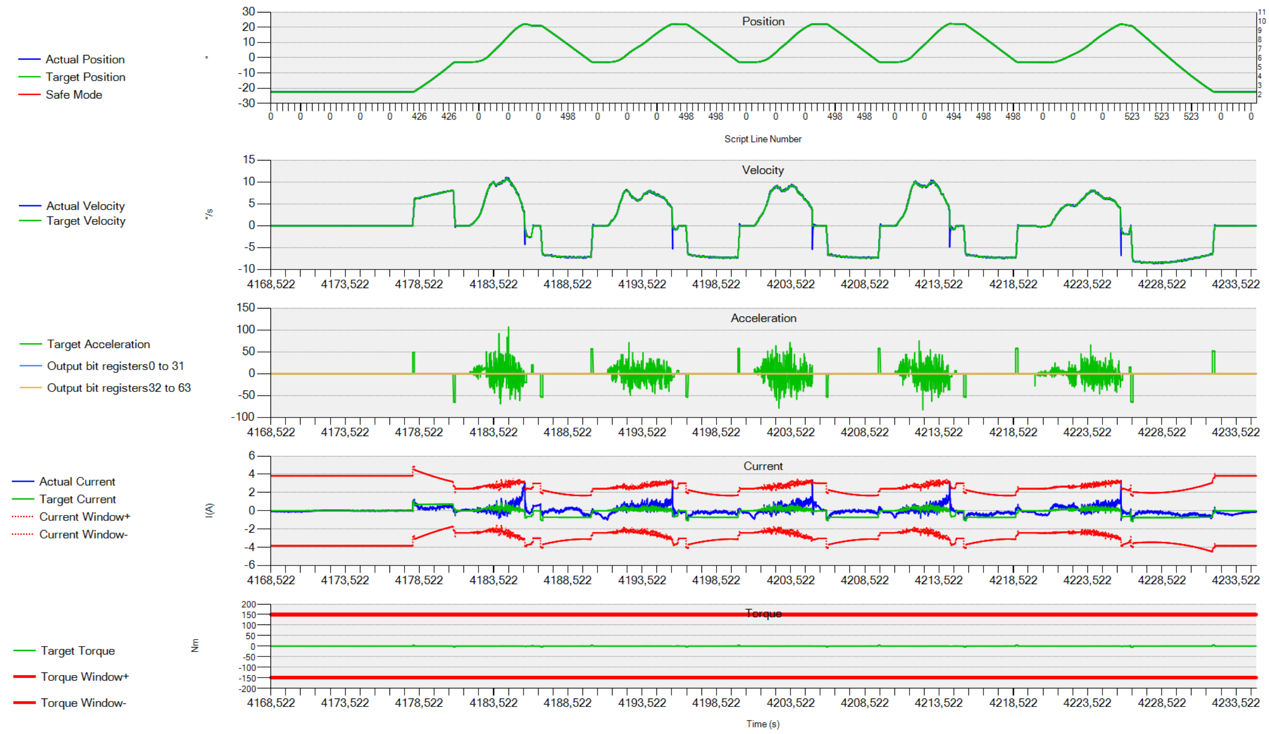


Figure 118 - Healthy subject 12. Left Side. D-Modality.

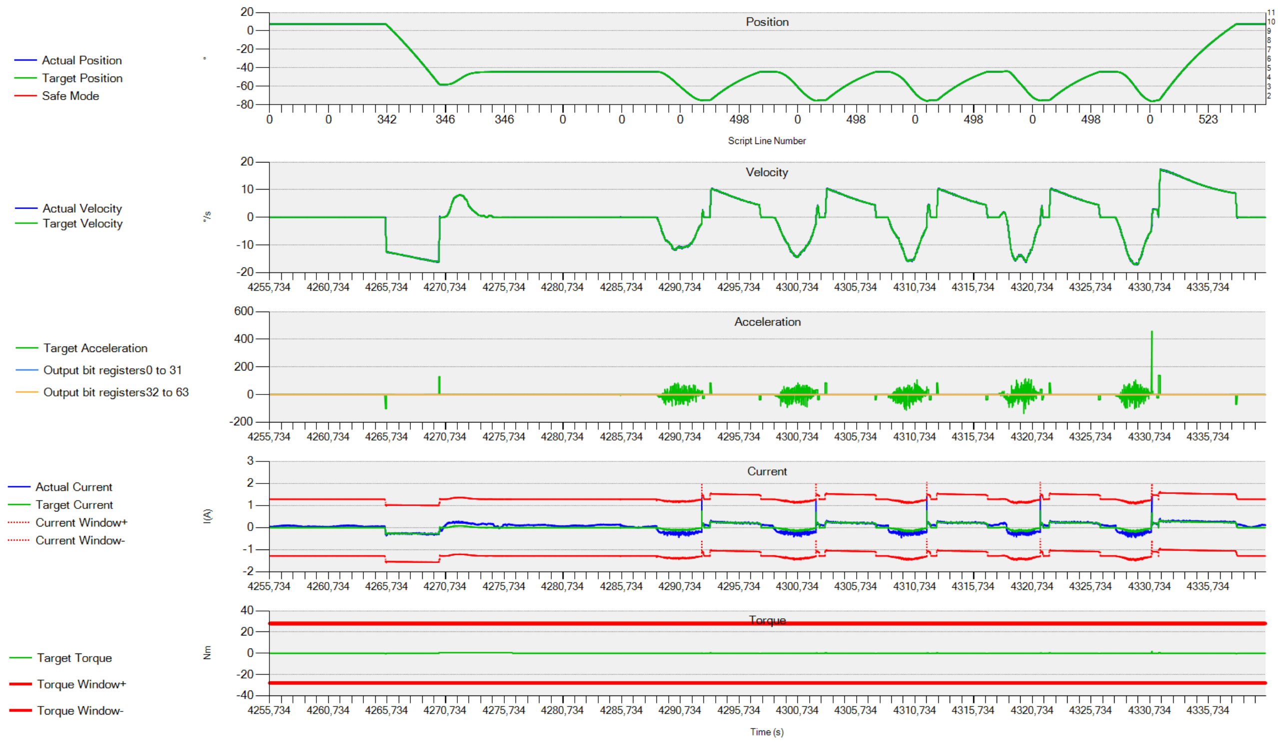


Figure 119 - Healthy subject 12. Right Side. F-Modality.

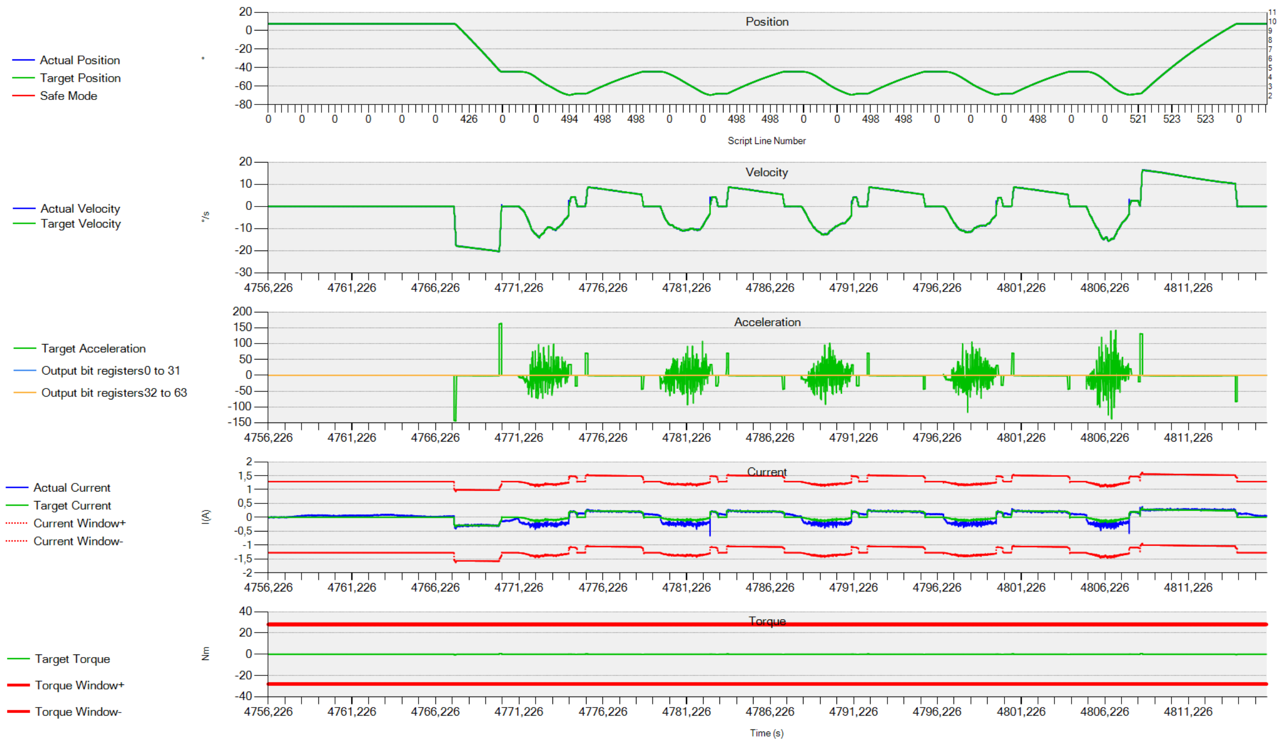


Figure 120 - Healthy subject 12. Right Side. M-Modality.

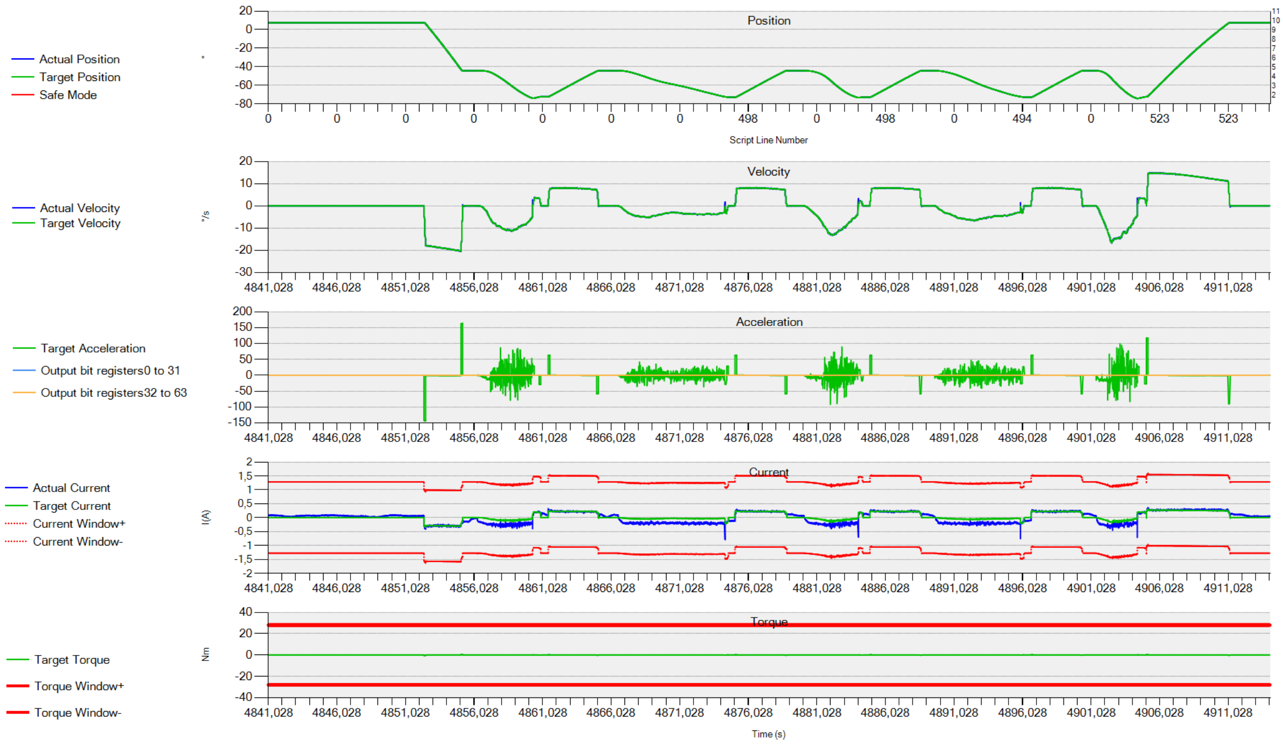


Figure 121 - Healthy subject 12. Right Side. D-Modality.

HEALTHY SUBJECT 13

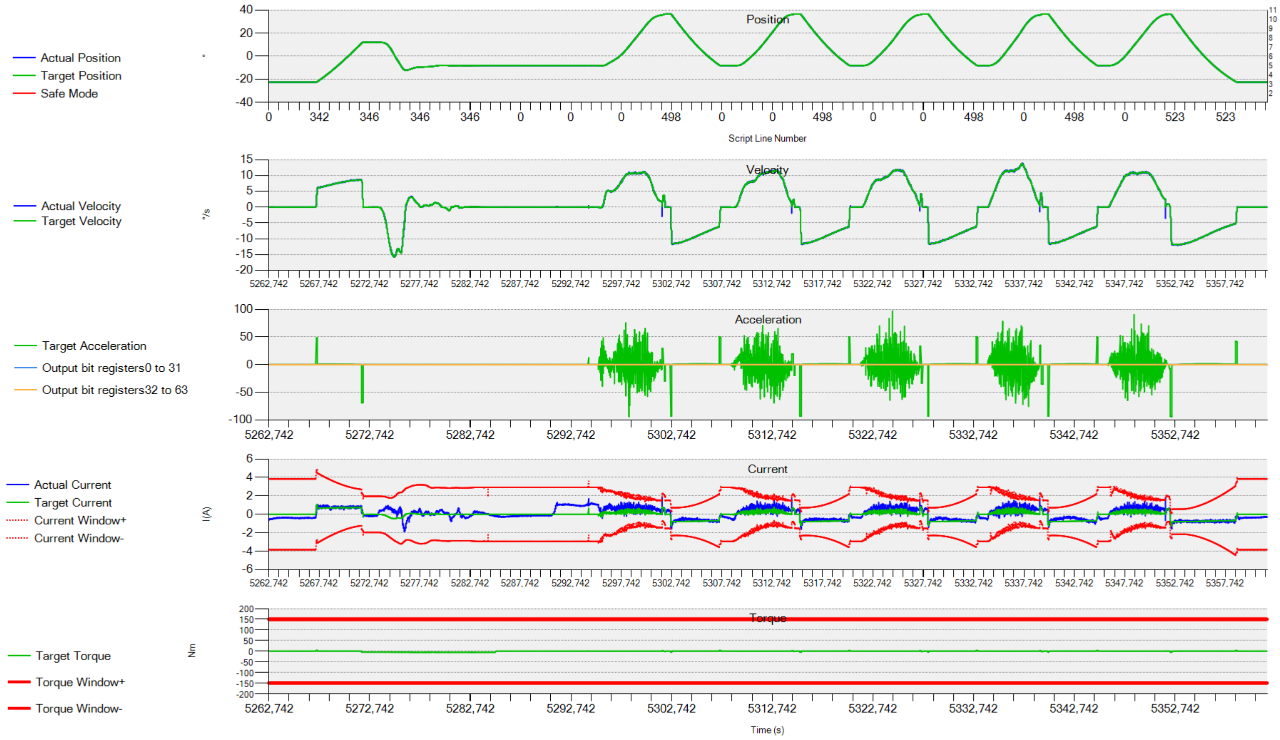


Figure 122 - Healthy subject 13. Left Side. F-Modality

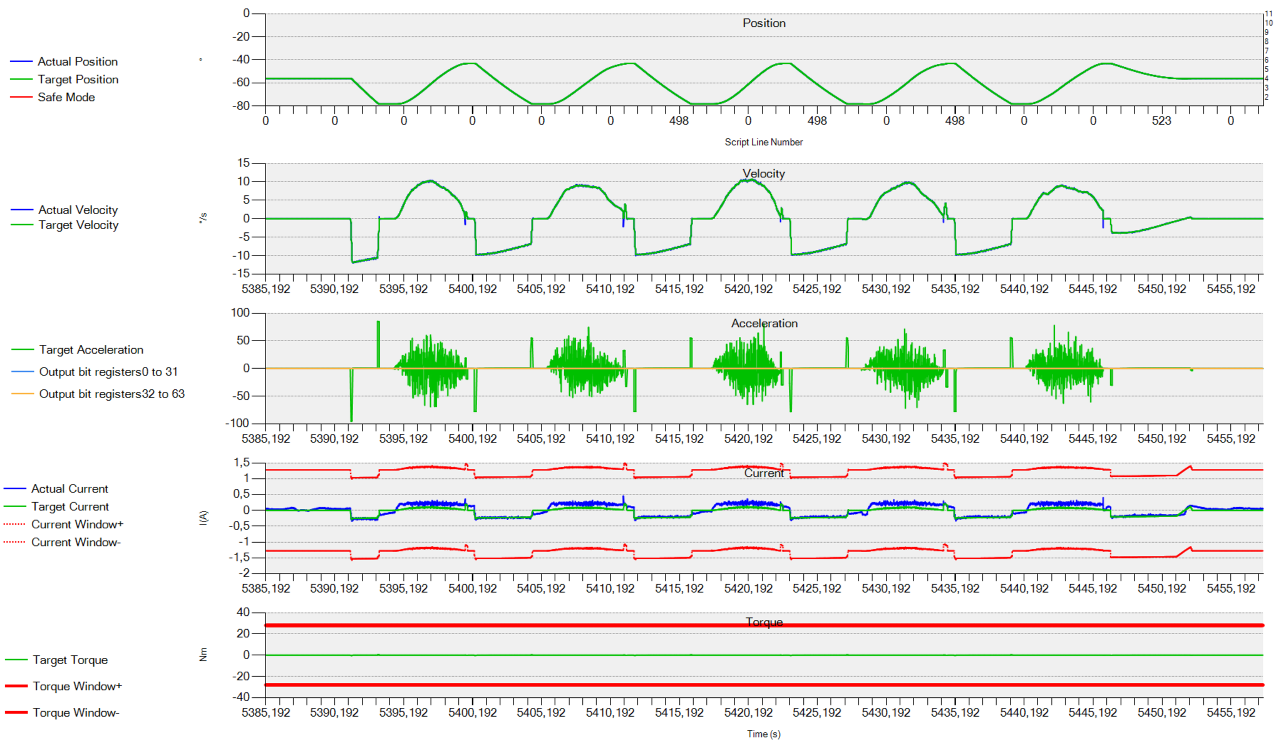


Figure 123 - Healthy subject 13. Left Side. M-Modality.

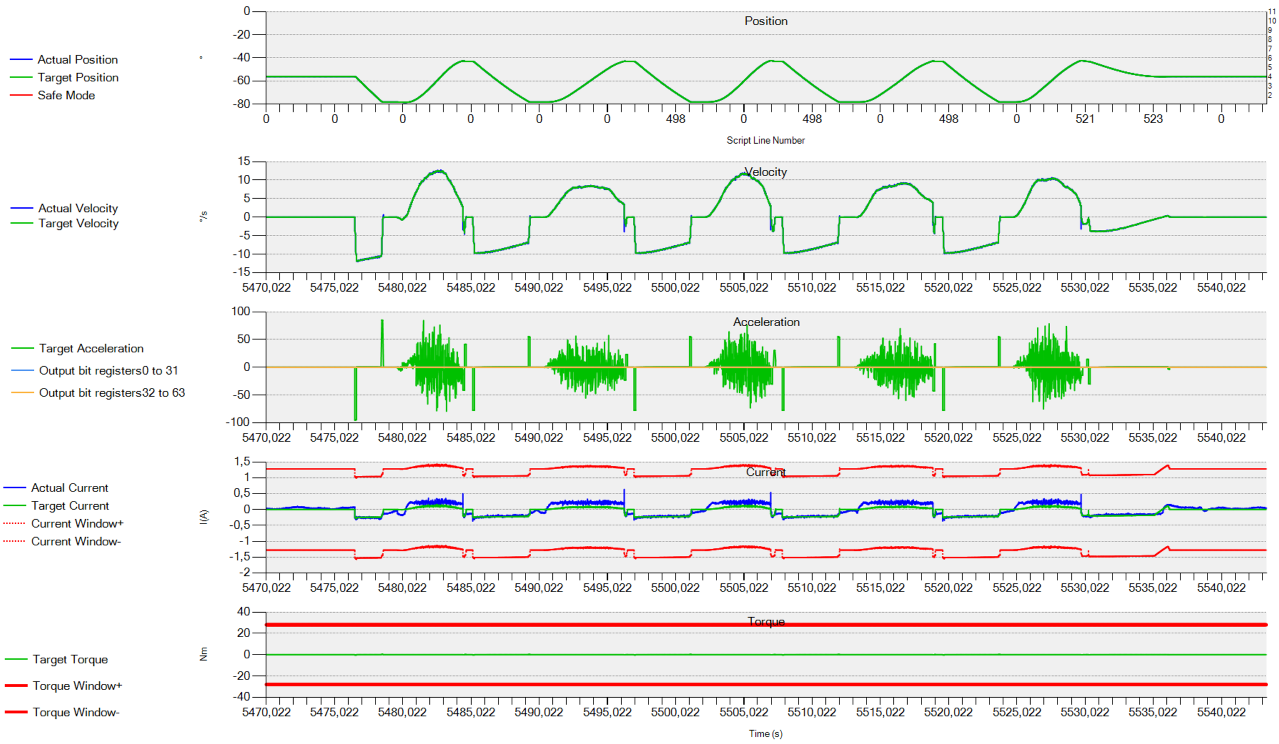


Figure 124- Healthy subject 13. Left Side. D-Modality.

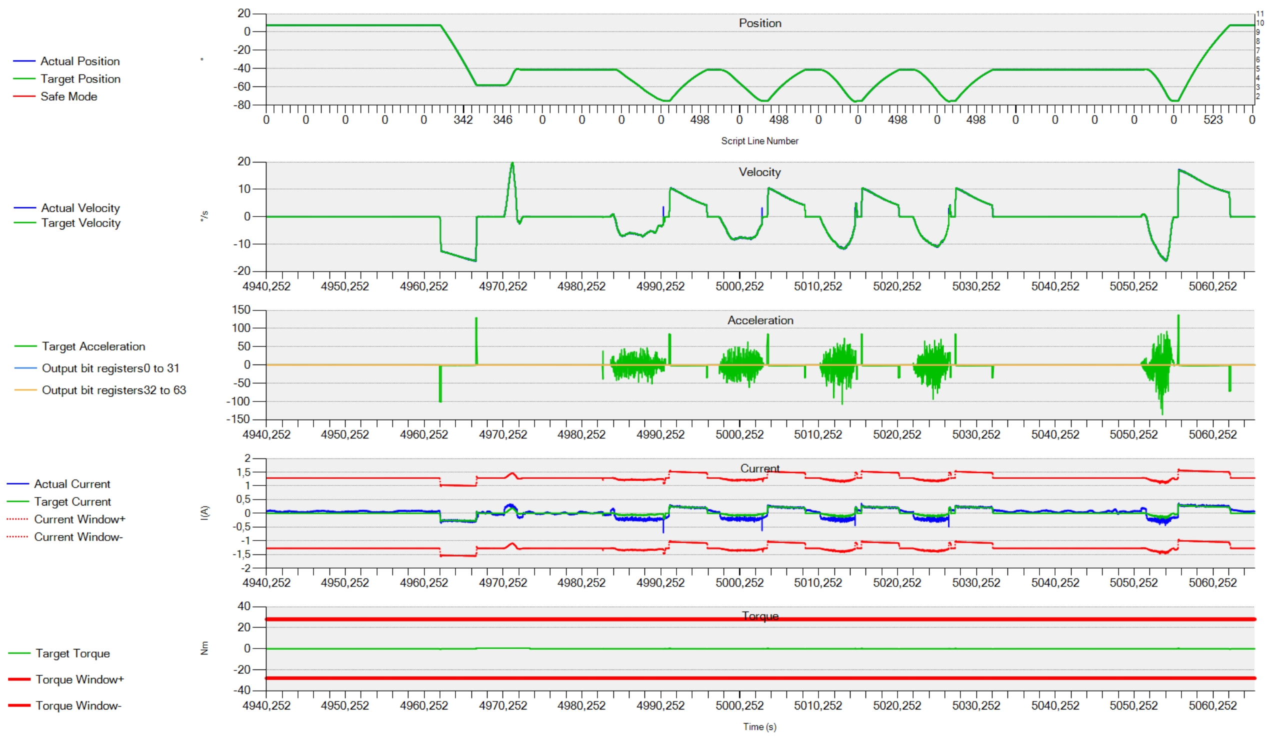


Figure 125 - Healthy subject 13. Right Side. F-Modality.

Figure 126 - Healthy subject 13. Right Side. M-Modality.

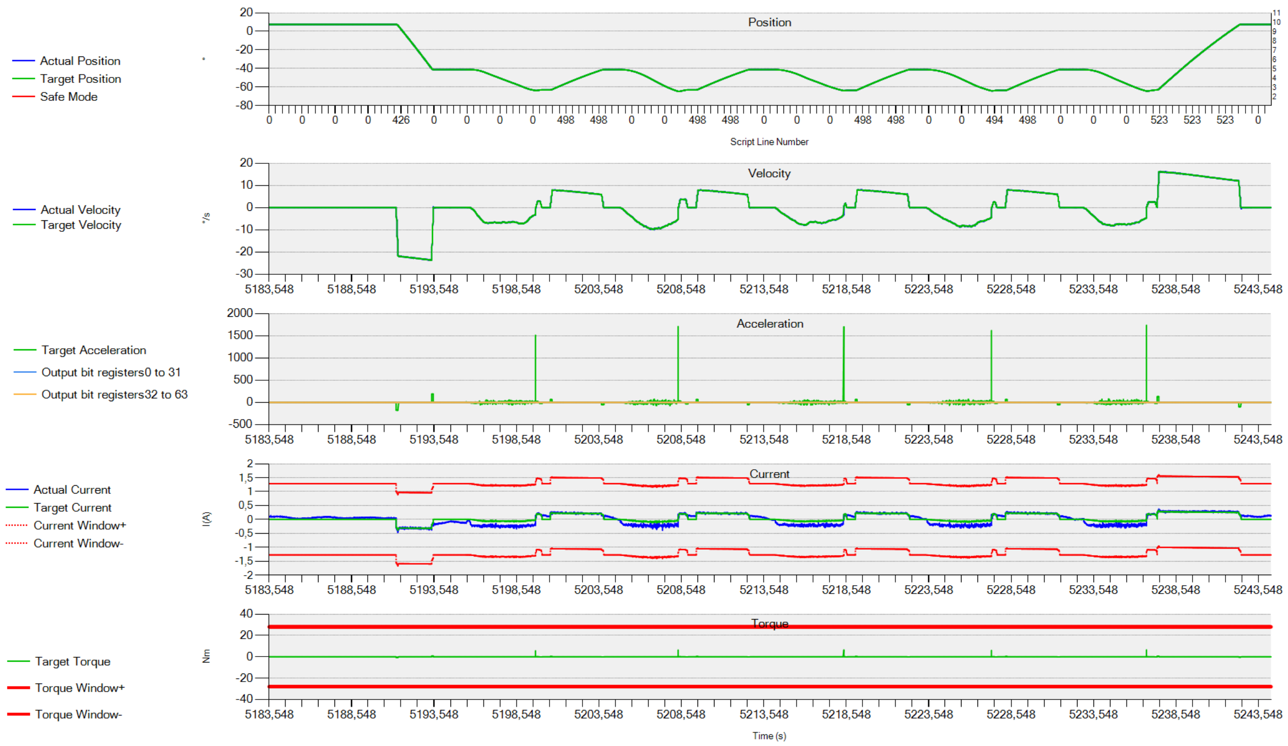


Figure 127 - Healthy subject 13. Right Side. D-Modality.

HEALTHY SUBJECT 14

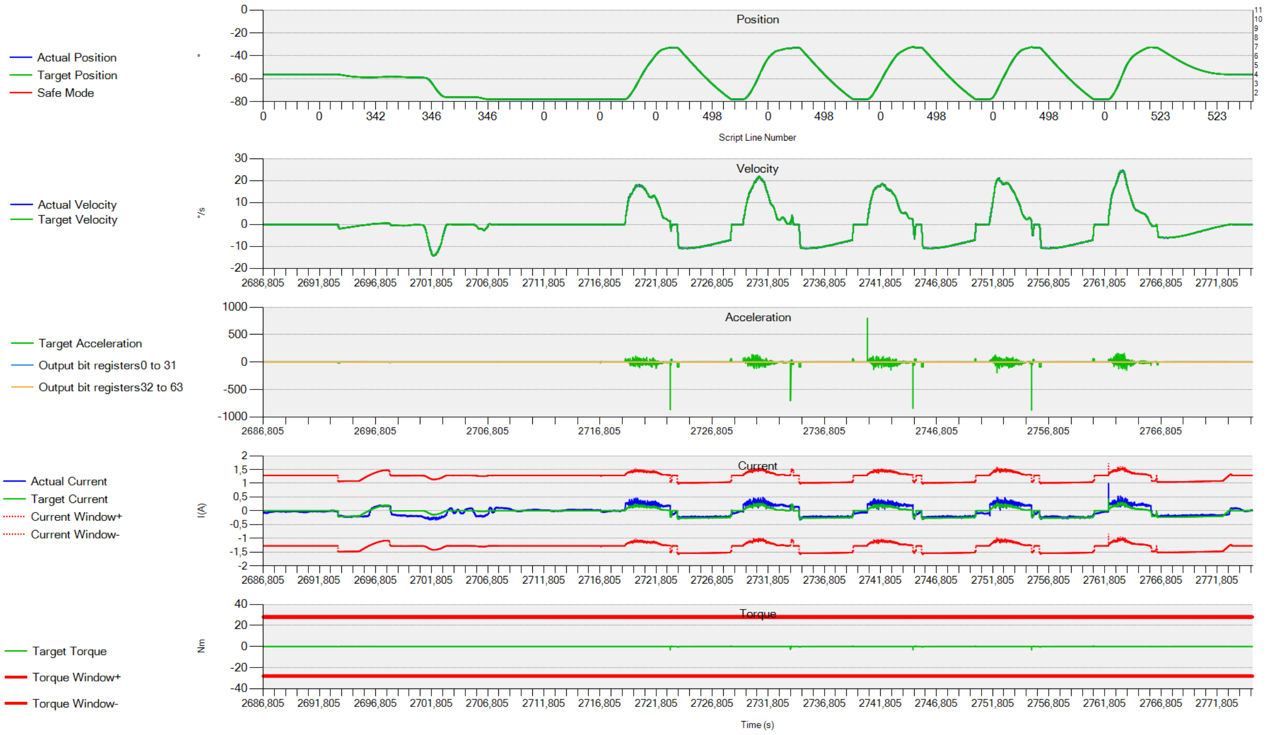


Figure 128 - Healthy subject 14. Left Side. F-Modality

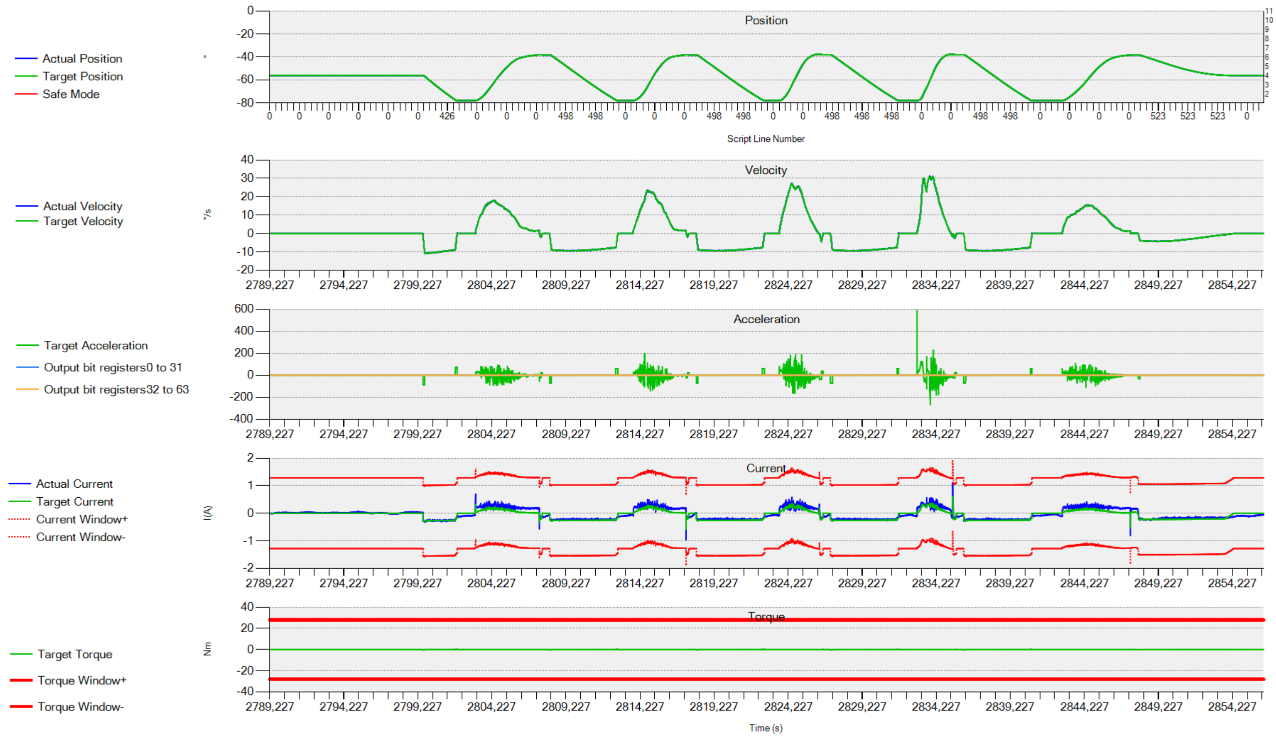


Figure 129 - Healthy subject 14. Left Side. M-Modality.

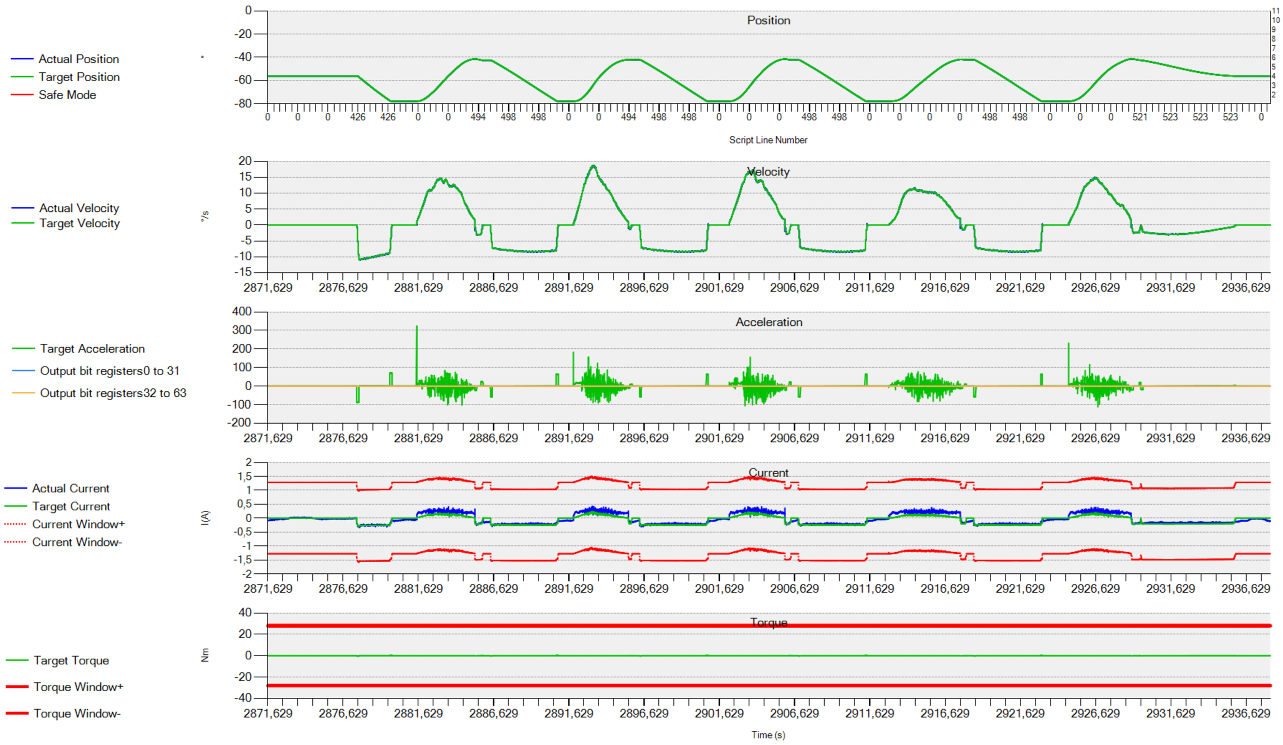


Figure 130 - Healthy subject 14. Left Side. D-Modality.

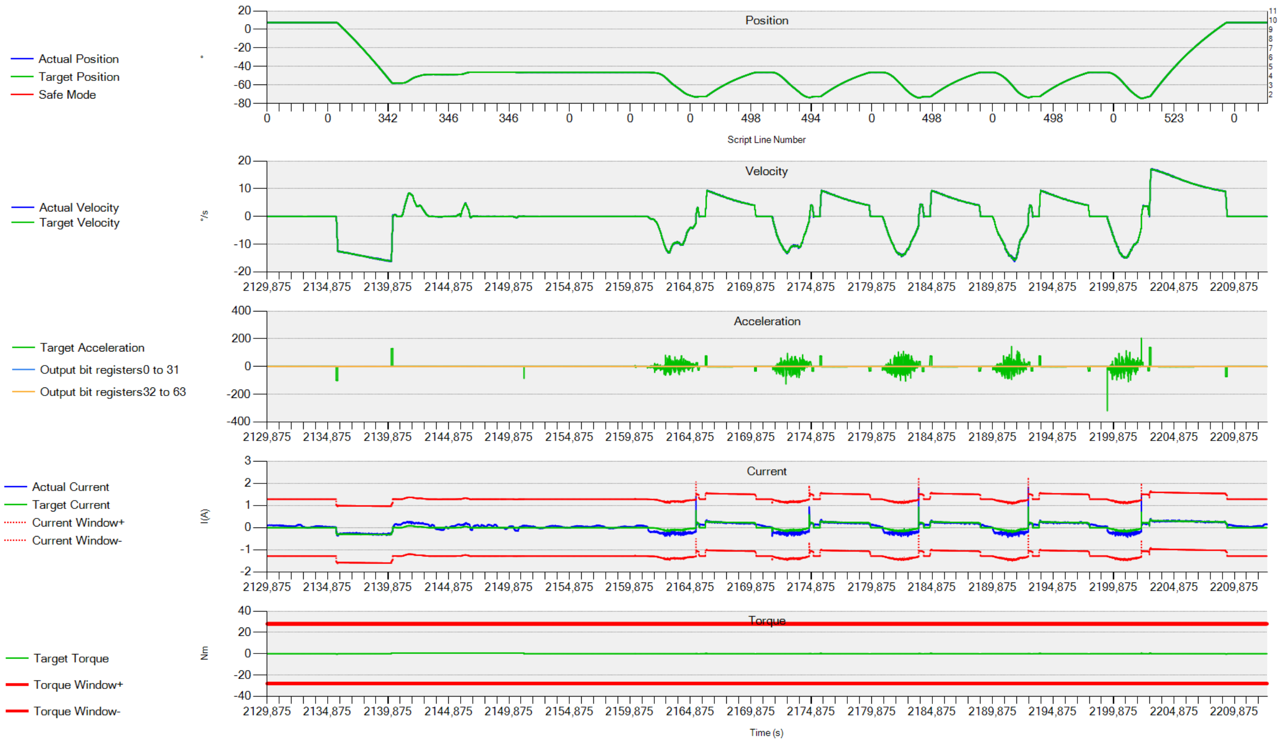


Figure 131 - Healthy subject 14. Right Side. F-Modality.

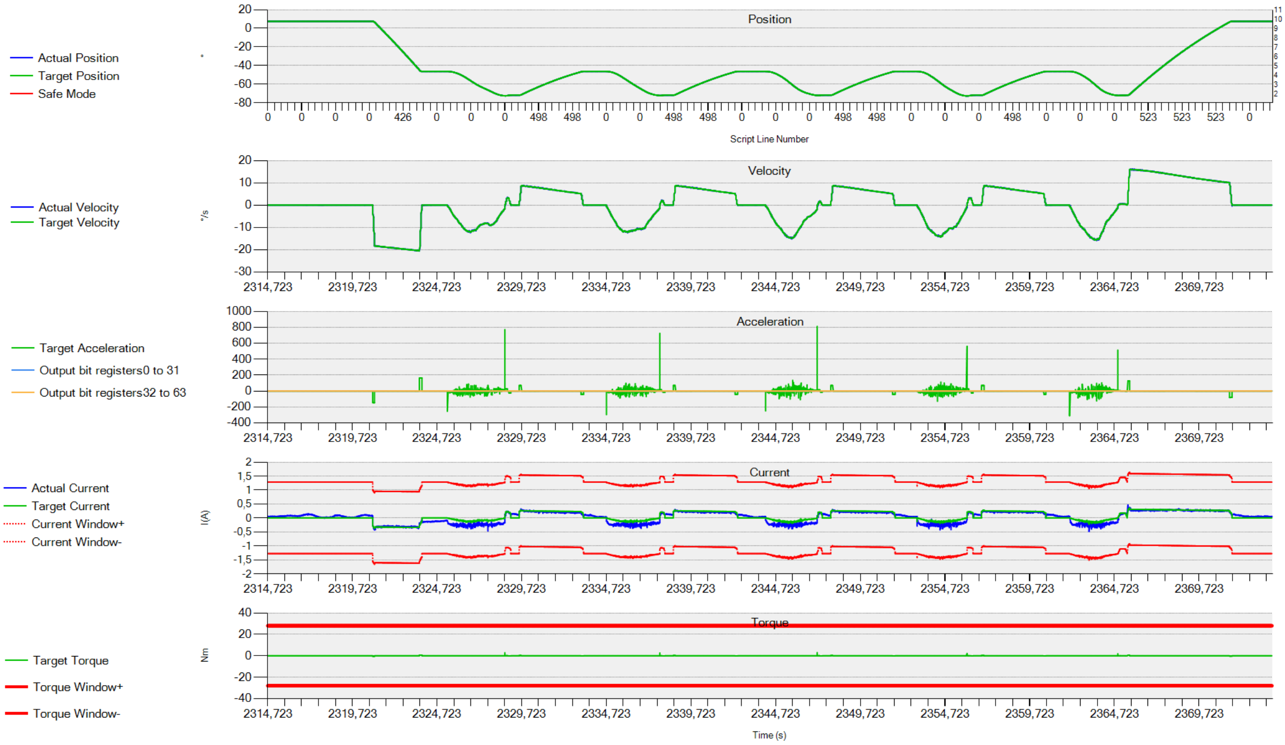


Figure 132 - Healthy subject 14. Right Side. M-Modality.

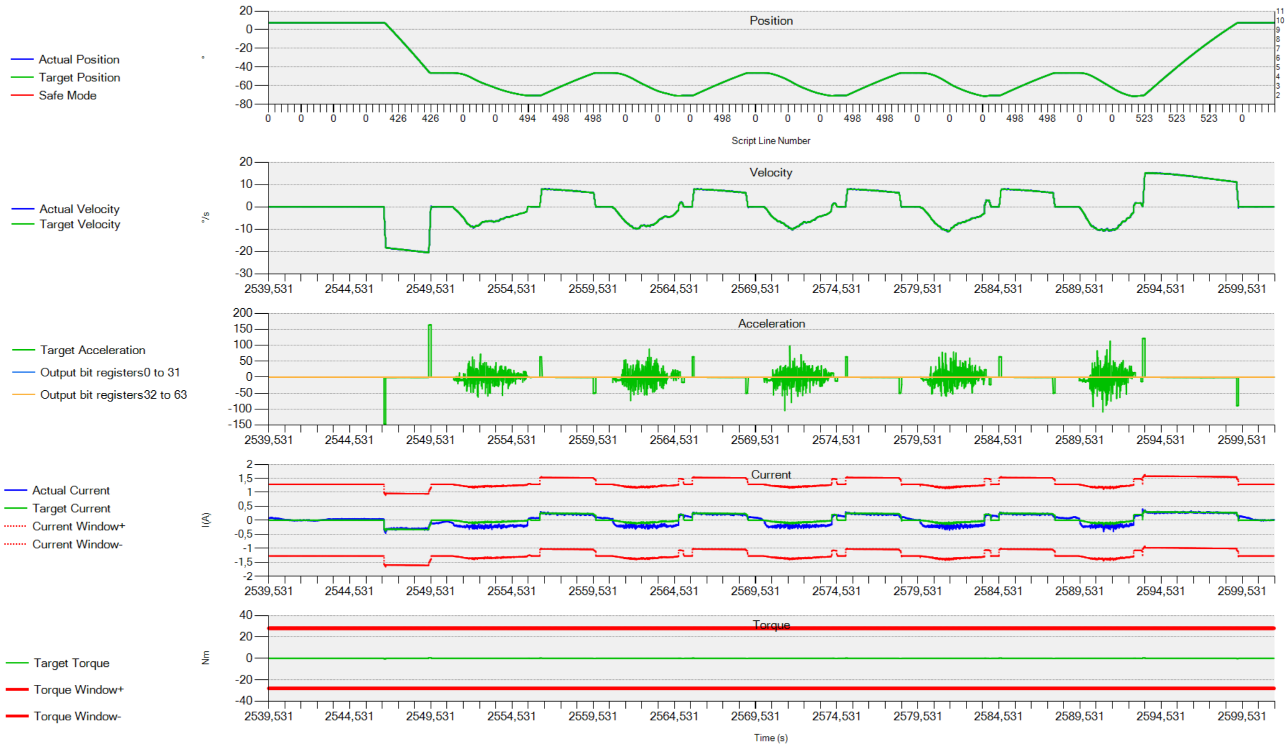


Figure 133 - Healthy subject 14. Right Side. D-Modality.

HEALTHY SUBJECT 15

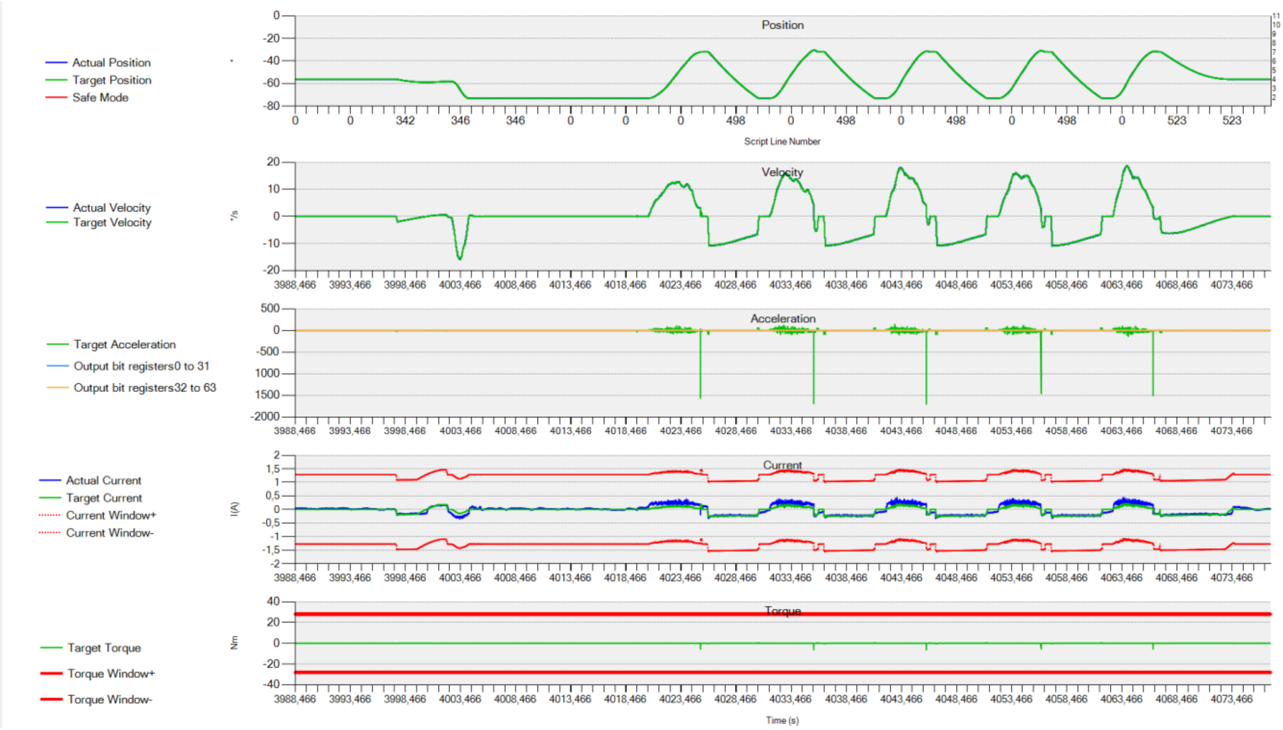


Figure 134 - Healthy subject 15. Left Side. F-Modality

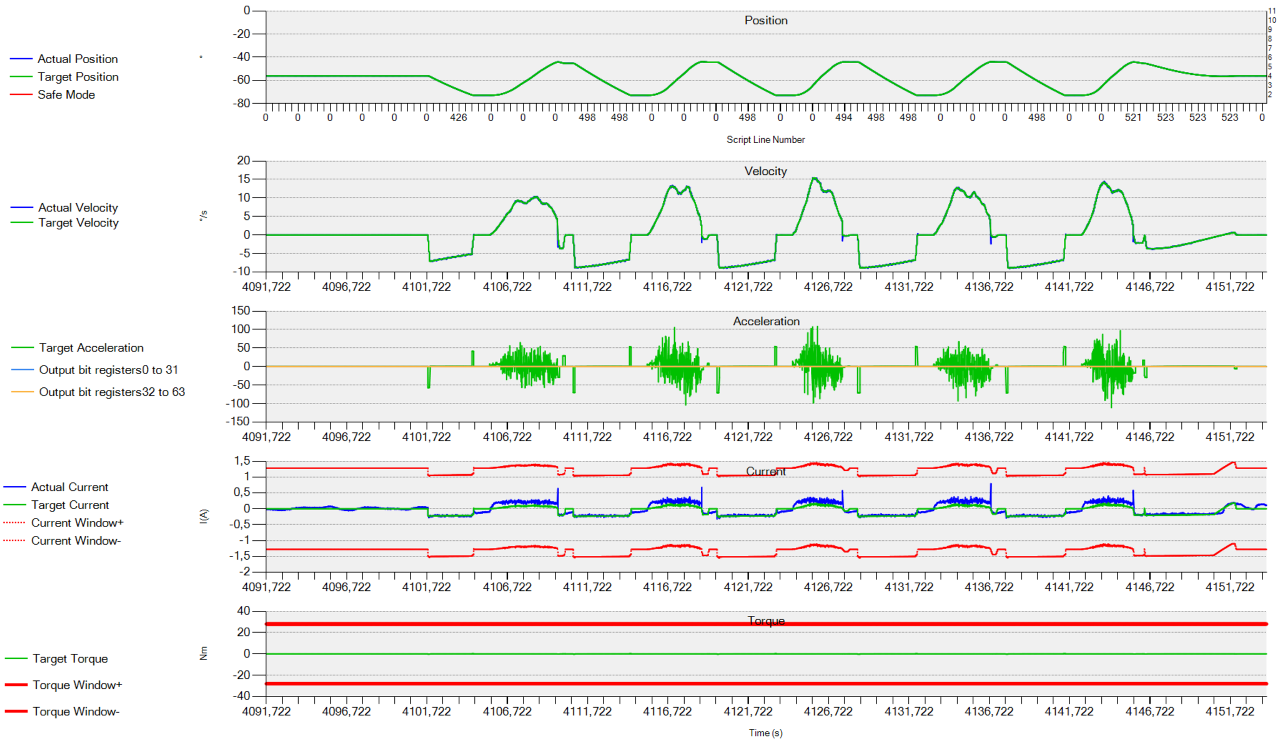


Figure 135 - Healthy subject 15. Left Side. M-Modality.

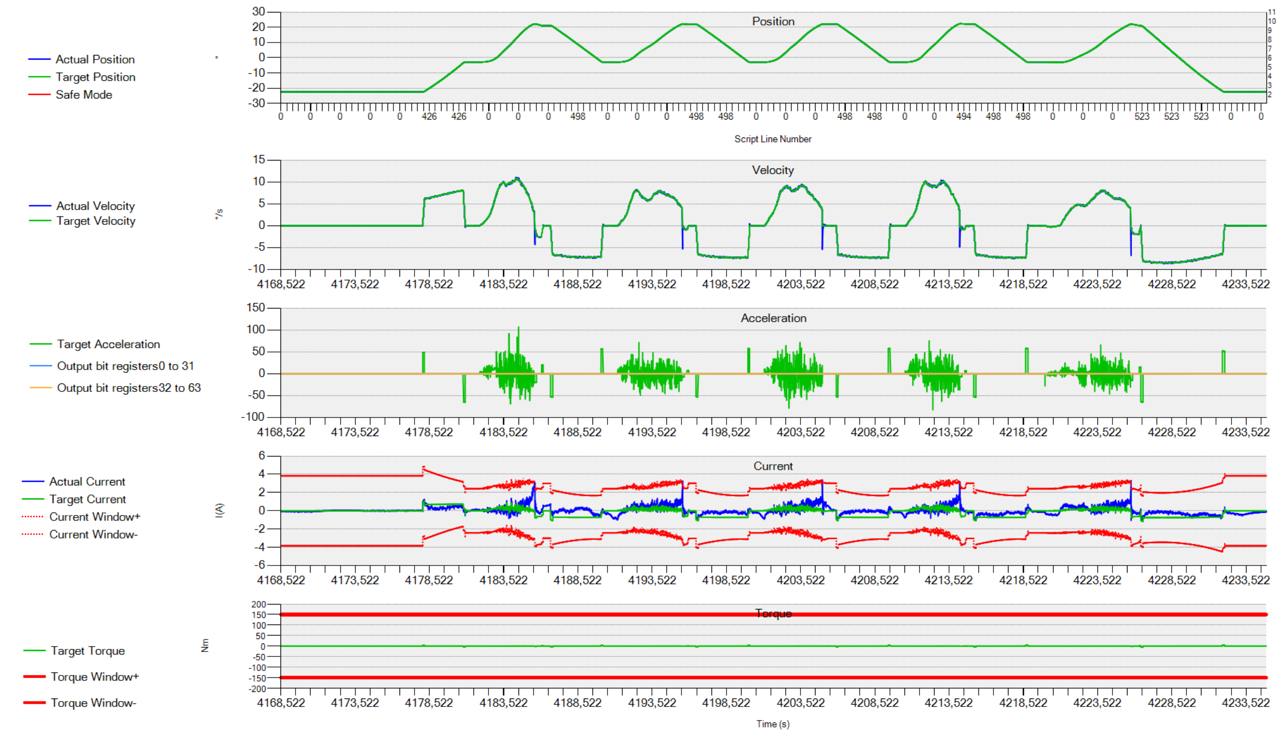


Figure 136 - Healthy subject 15. Left Side. D-Modality.

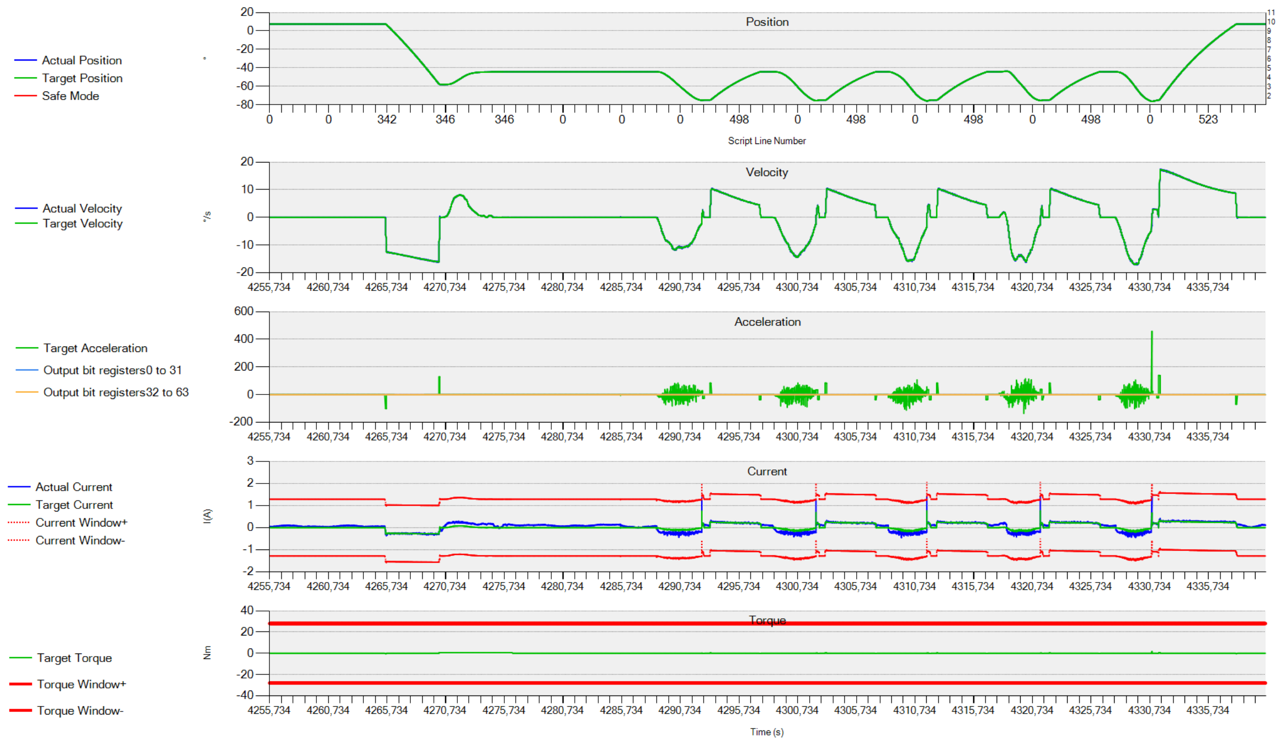


Figure 137 - Healthy subject 15. Right Side. F-Modality.

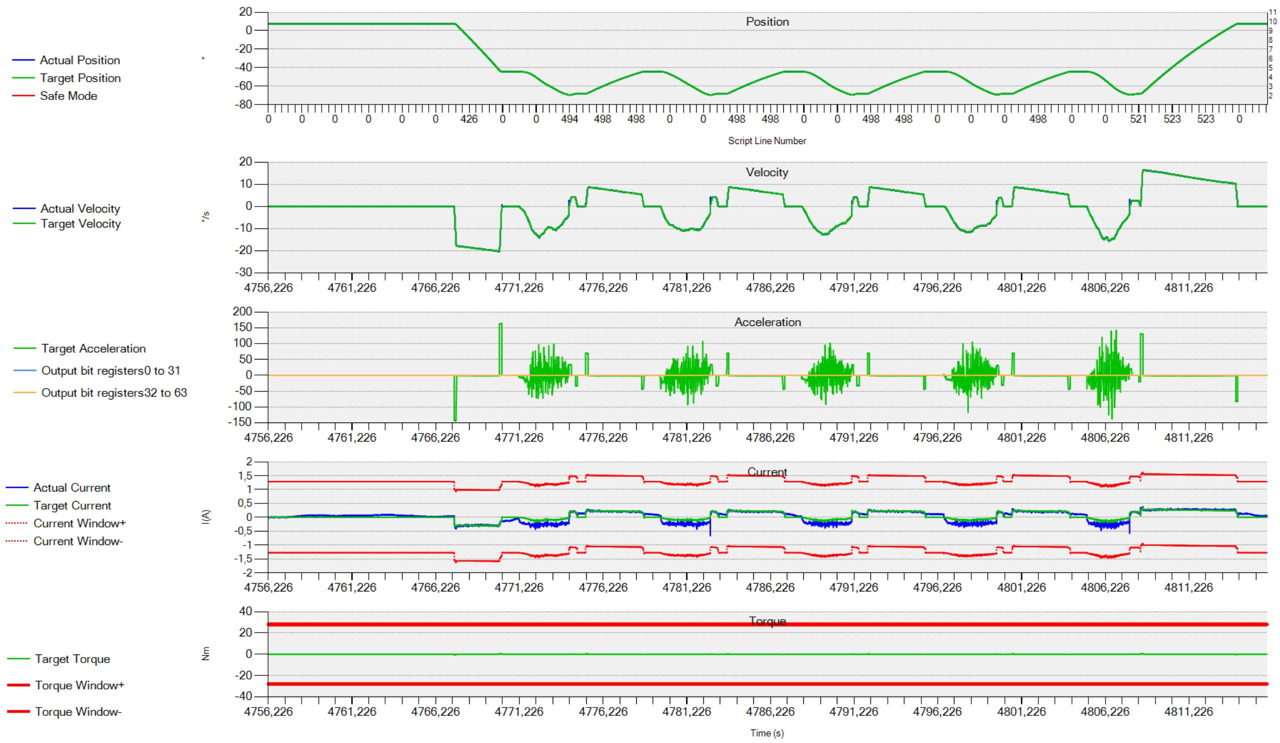


Figure 138 - Healthy subject 15. Right Side. M-Modality.

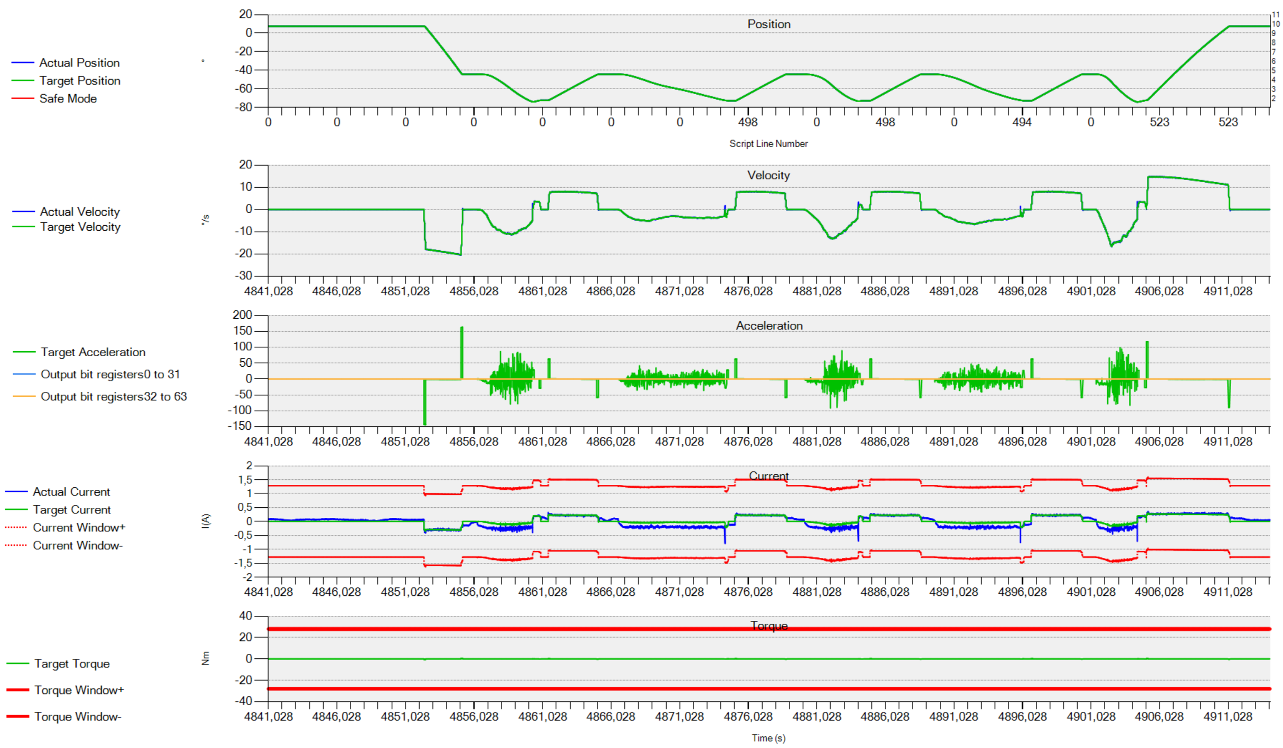


Figure 139 - Healthy subject 15. Right Side. D-Modality.

HEALTHY SUBJECT 16

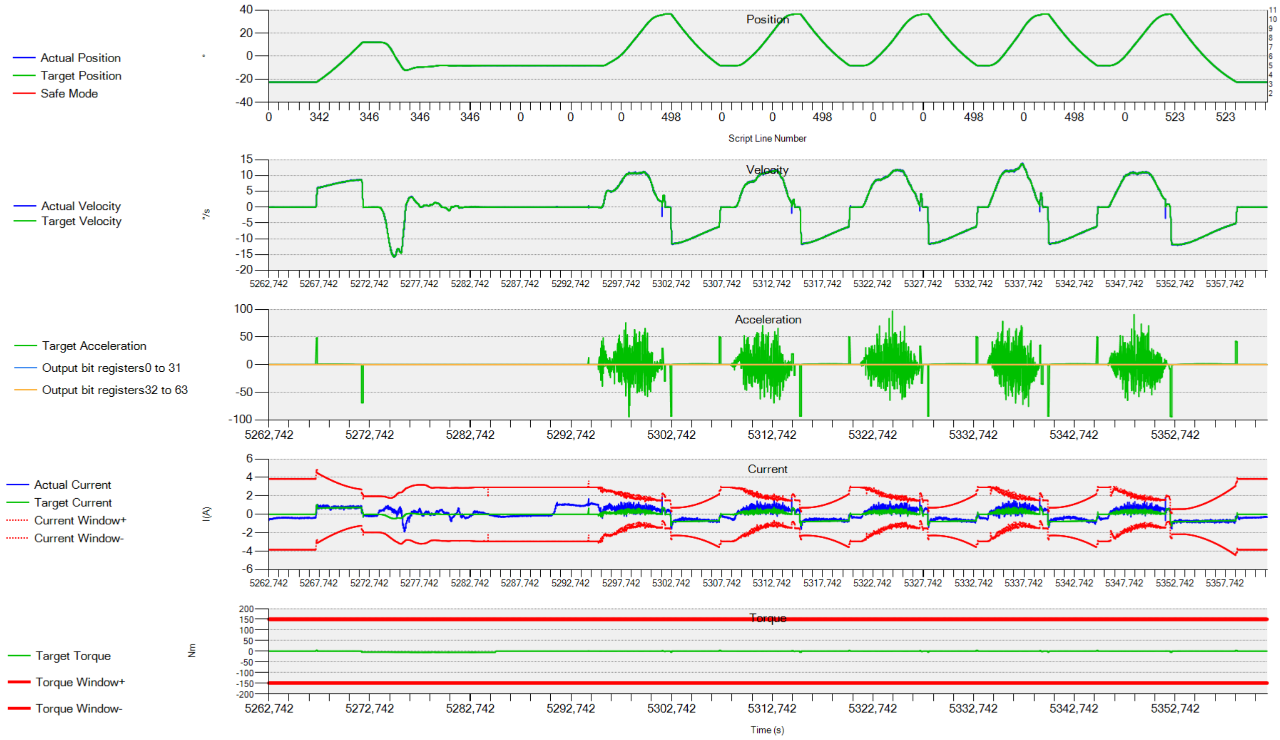


Figure 140 - Healthy subject 16. Left Side. F-Modality

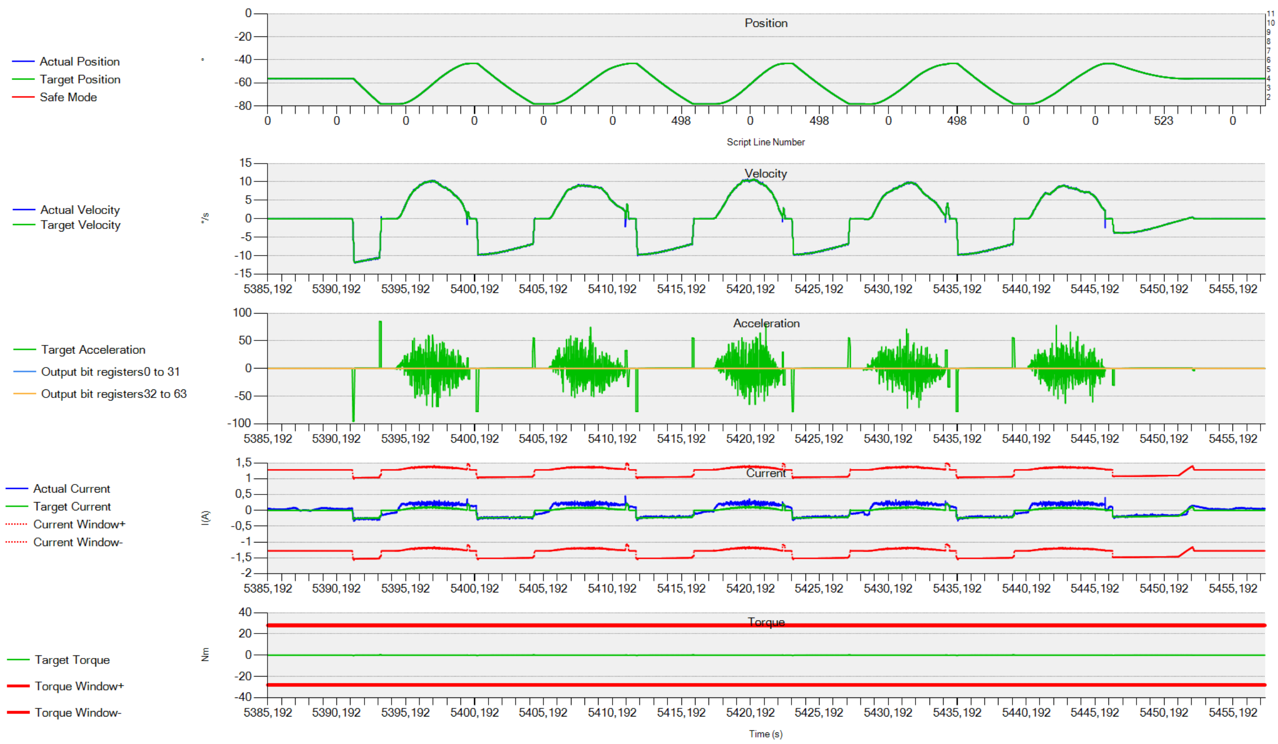


Figure 141 - Healthy subject 16. Left Side. M-Modality.

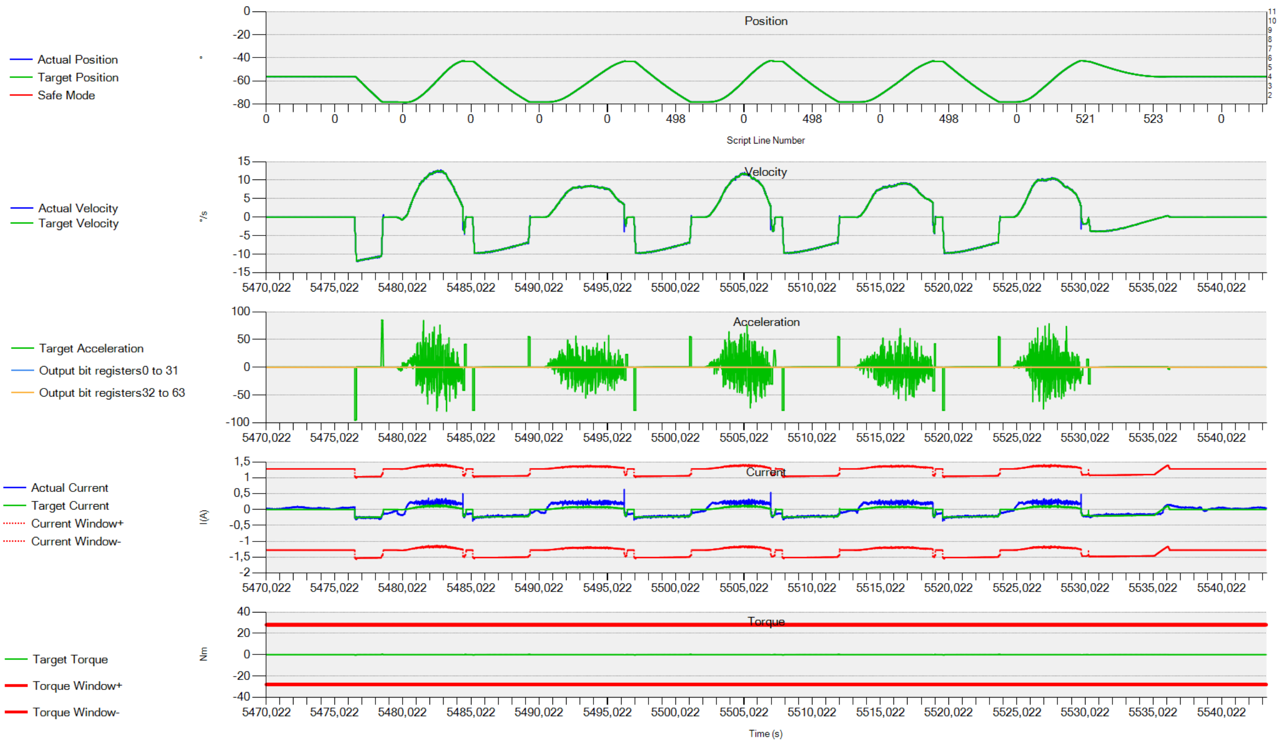


Figure 142- Healthy subject 16. Left Side. D-Modality.

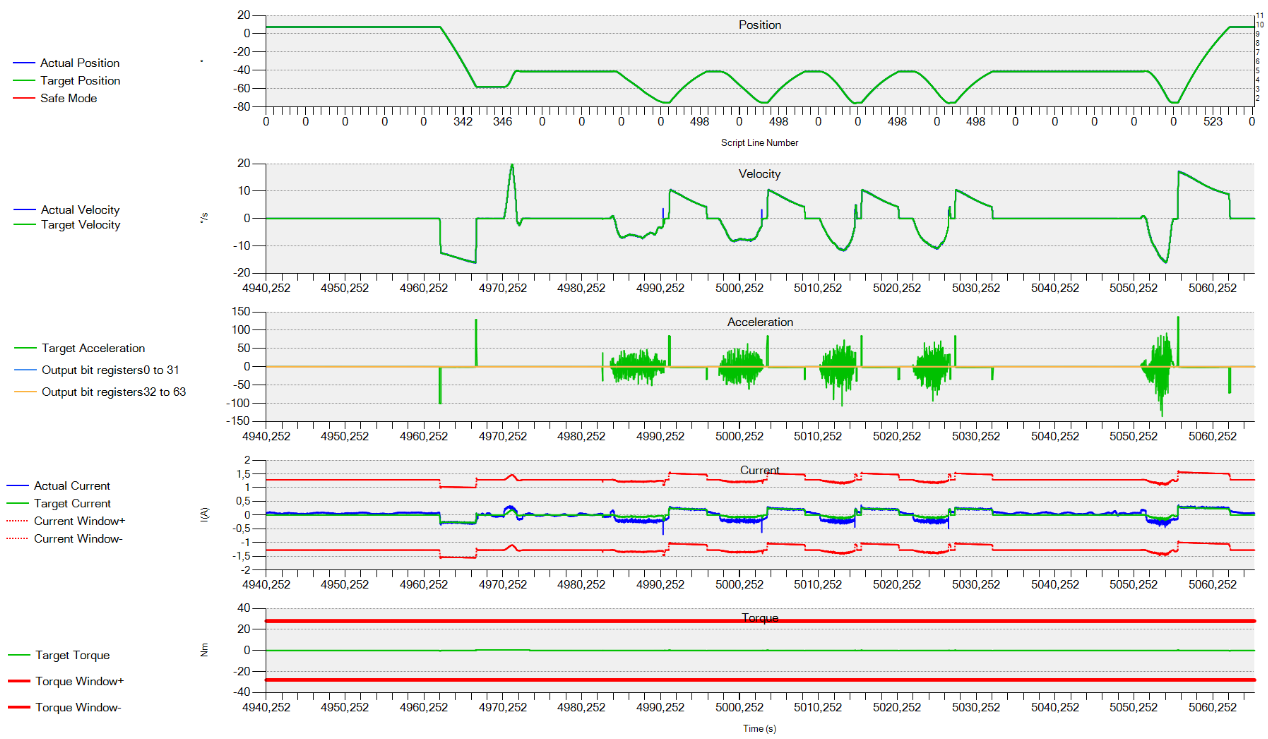


Figure 143 - Healthy subject 16. Right Side. F-Modality.

Figure 144 - Healthy subject 16. Right Side. M-Modality.

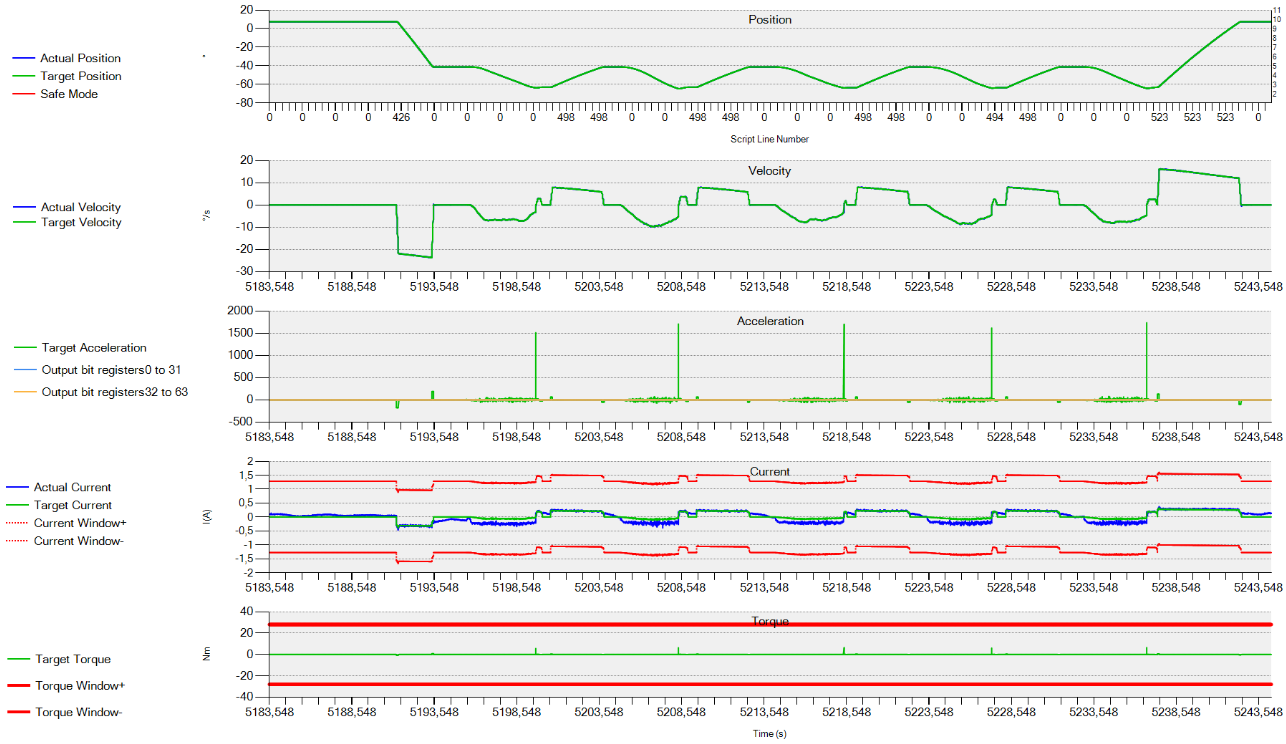


Figure 145 - Healthy subject 16. Right Side. D-Modality.

HEALTHY SUBJECT 17

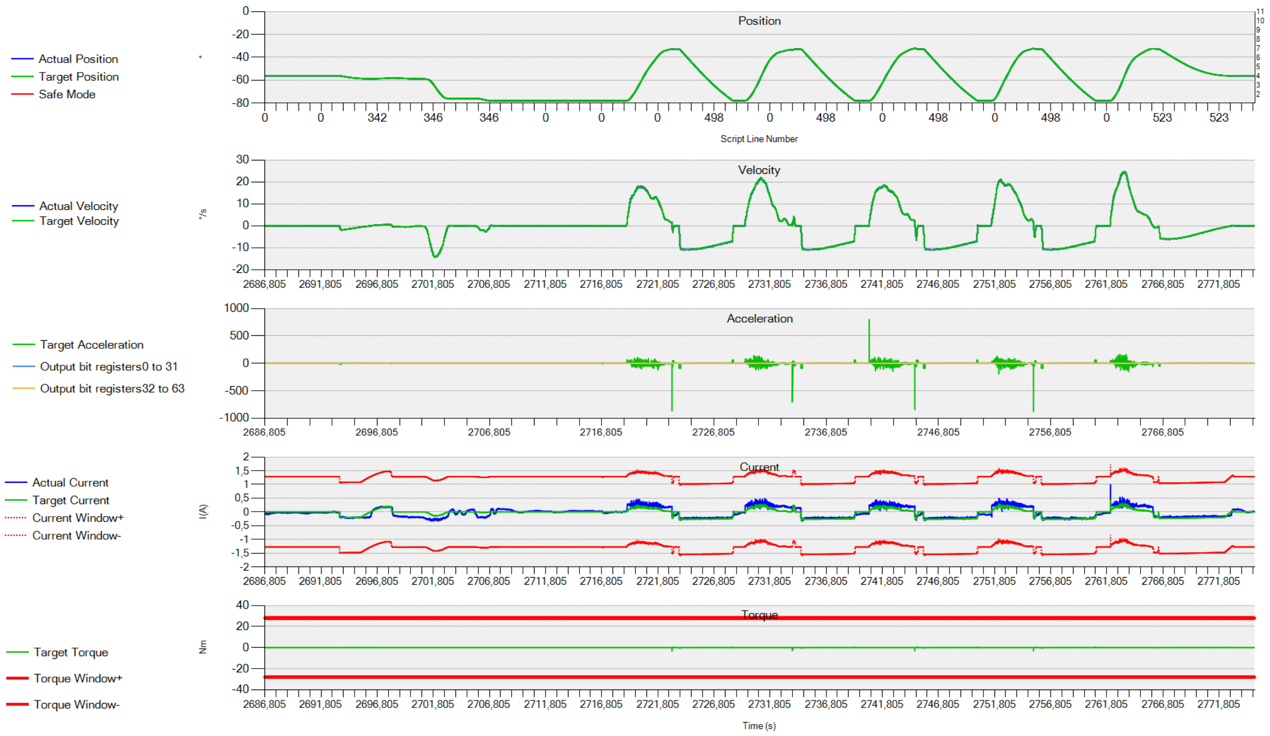


Figure 146 - Healthy subject 17. Left Side. F-Modality

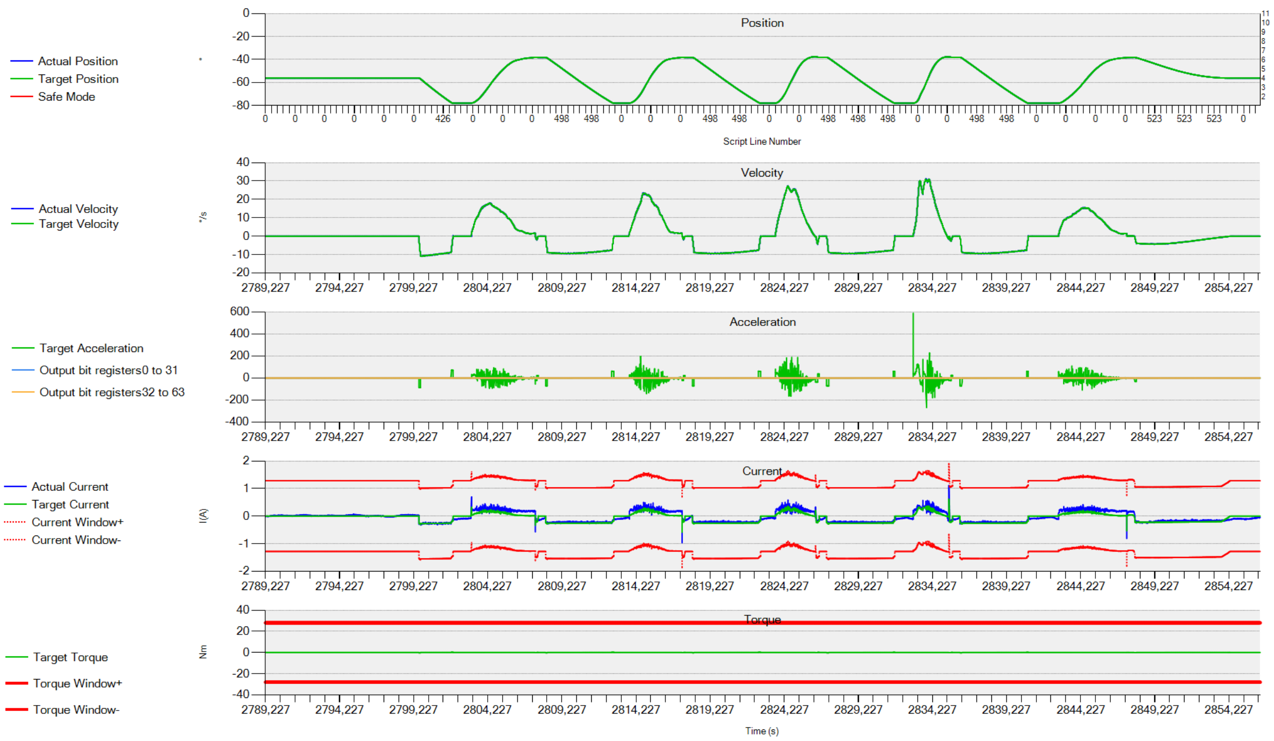


Figure 147 - Healthy subject 17. Left Side. M-Modality.

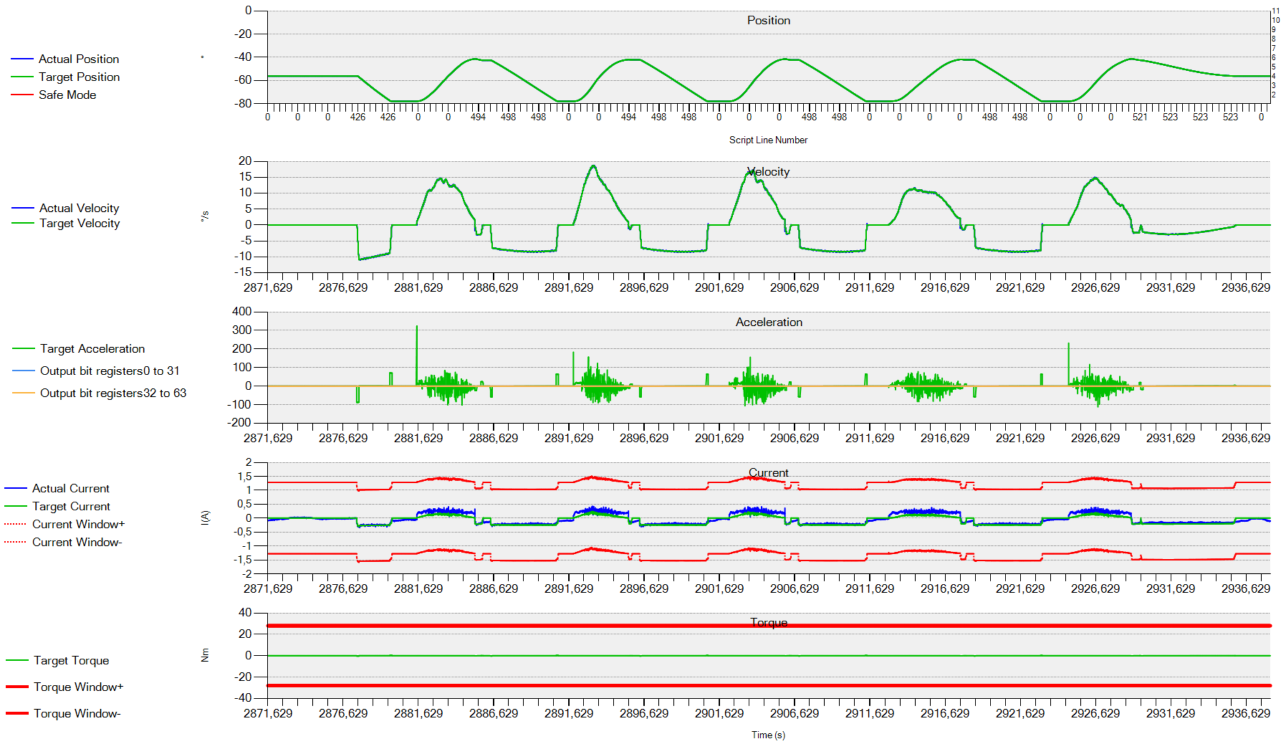


Figure 148 - Healthy subject 17. Left Side. D-Modality.

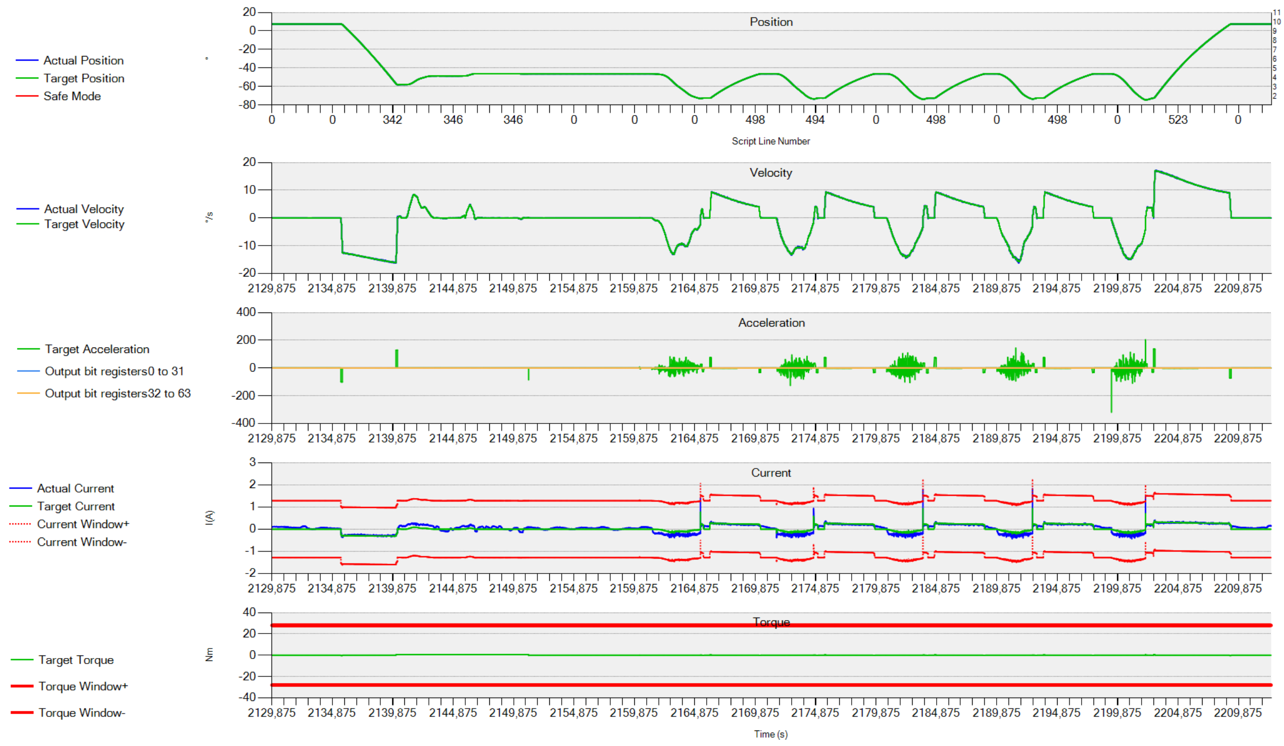


Figure 149 - Healthy subject 17. Right Side. F-Modality.

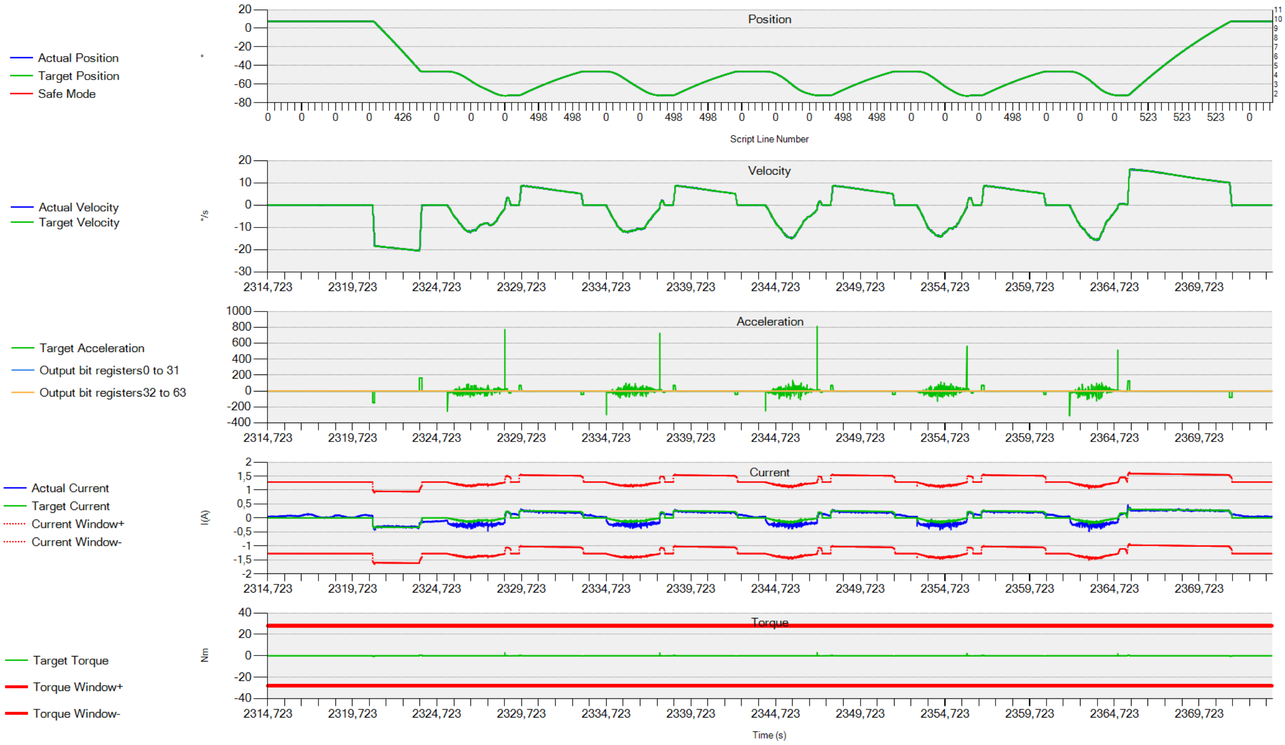


Figure 150 - Healthy subject 17. Right Side. M-Modality.

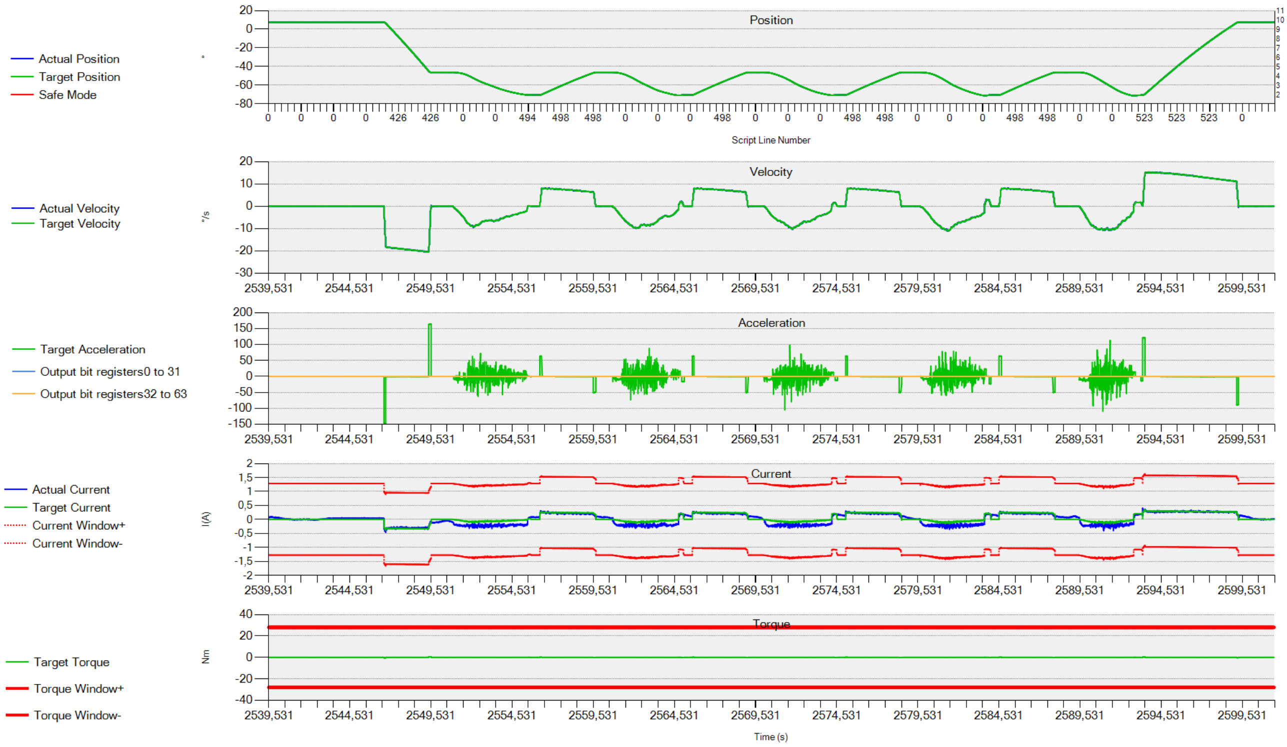


Figure 151 - Healthy subject 17. Right Side. D-Modality.

HEALTHY SUBJECT 18

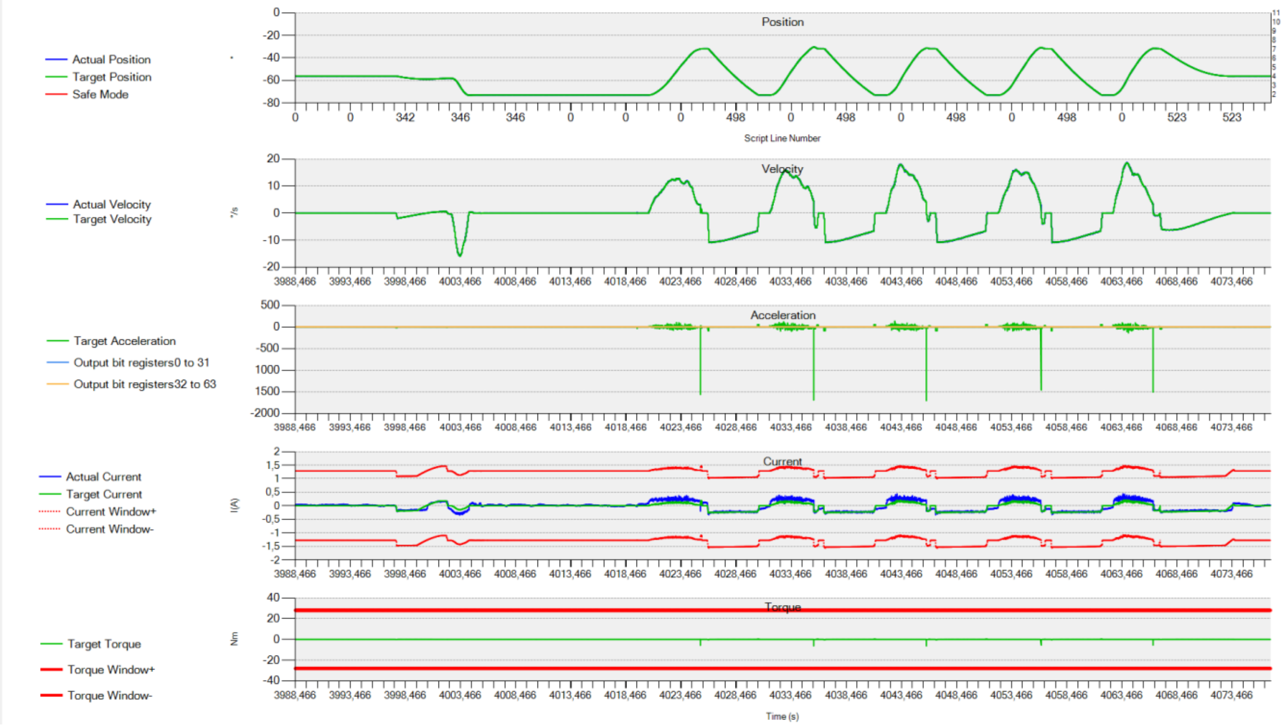


Figure 152 - Healthy subject 18. Left Side. F-Modality

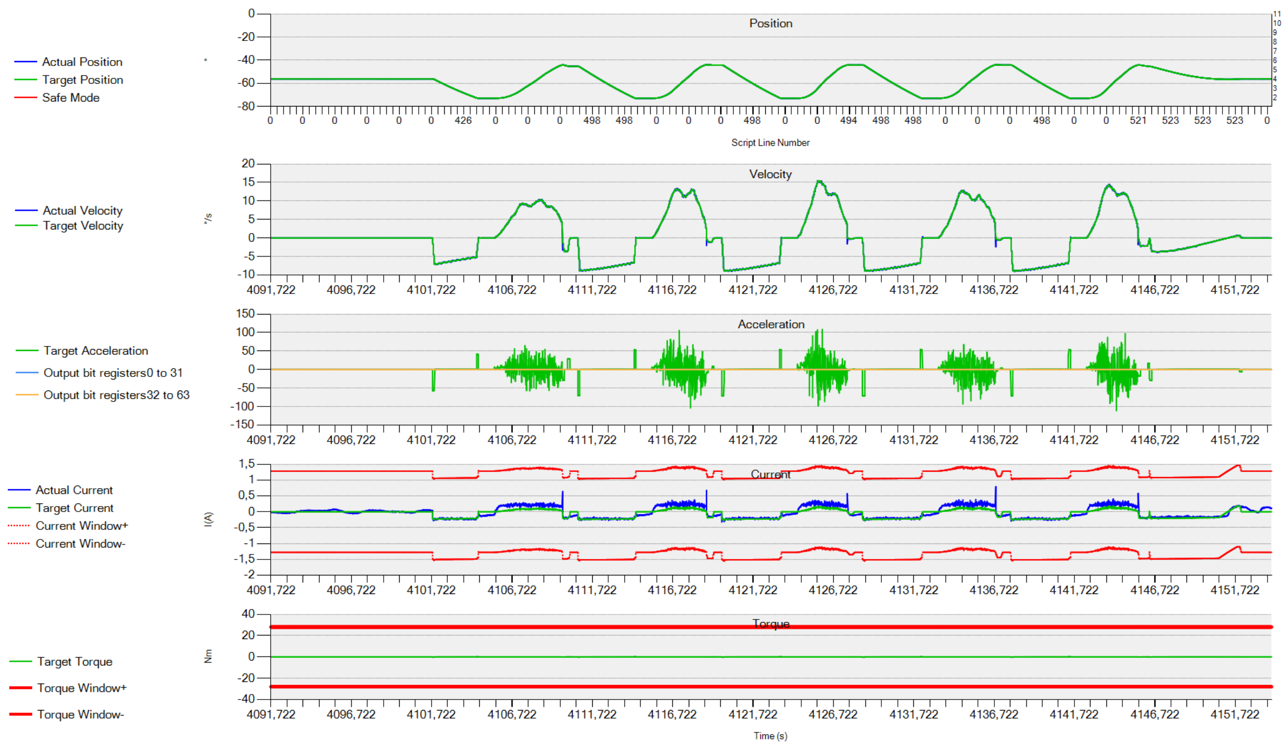


Figure 153 - Healthy subject 18. Left Side. M-Modality.

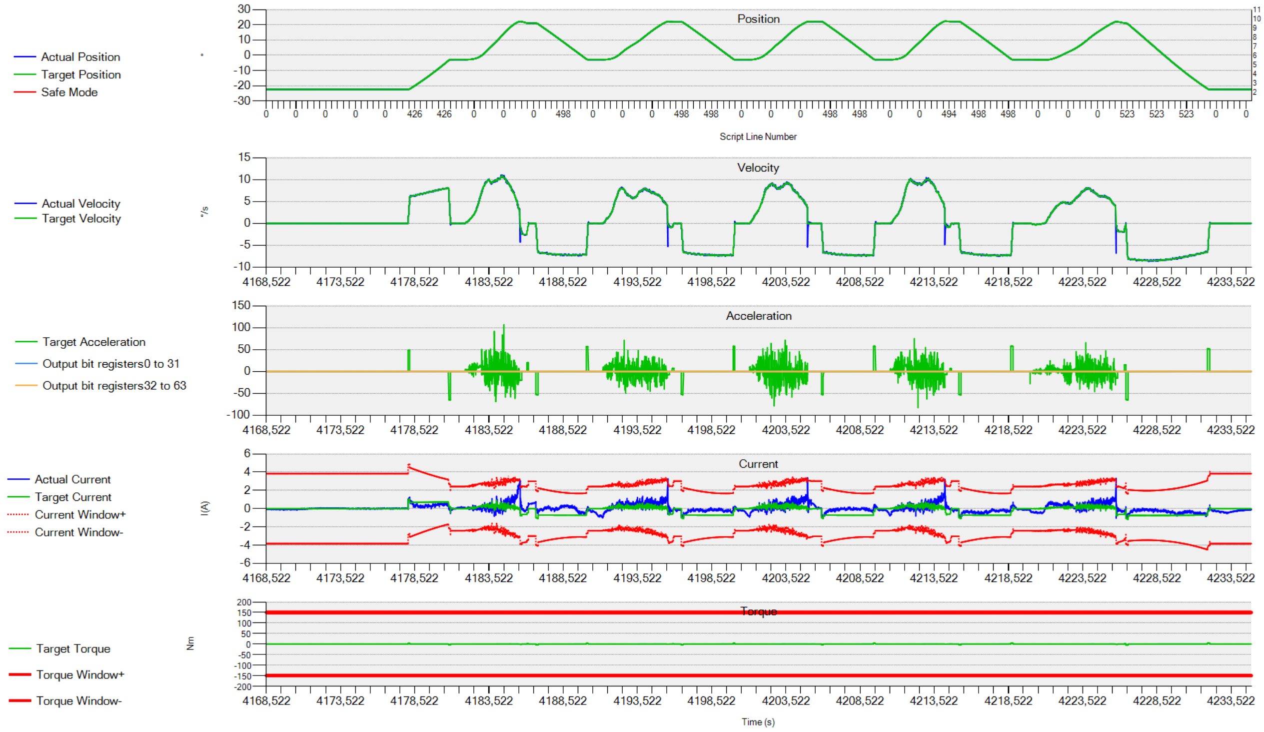


Figure 154 - Healthy subject 18. Left Side. D-Modality.

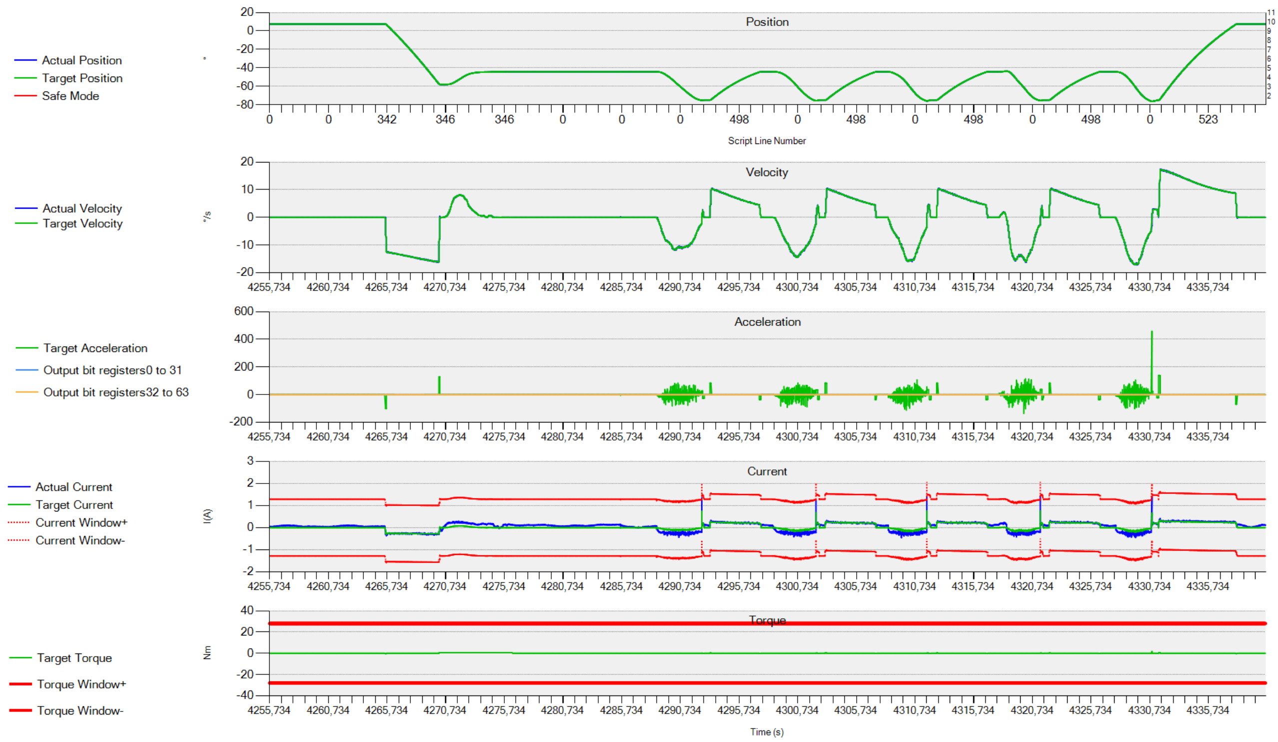


Figure 155 - Healthy subject 18. Right Side. F-Modality.

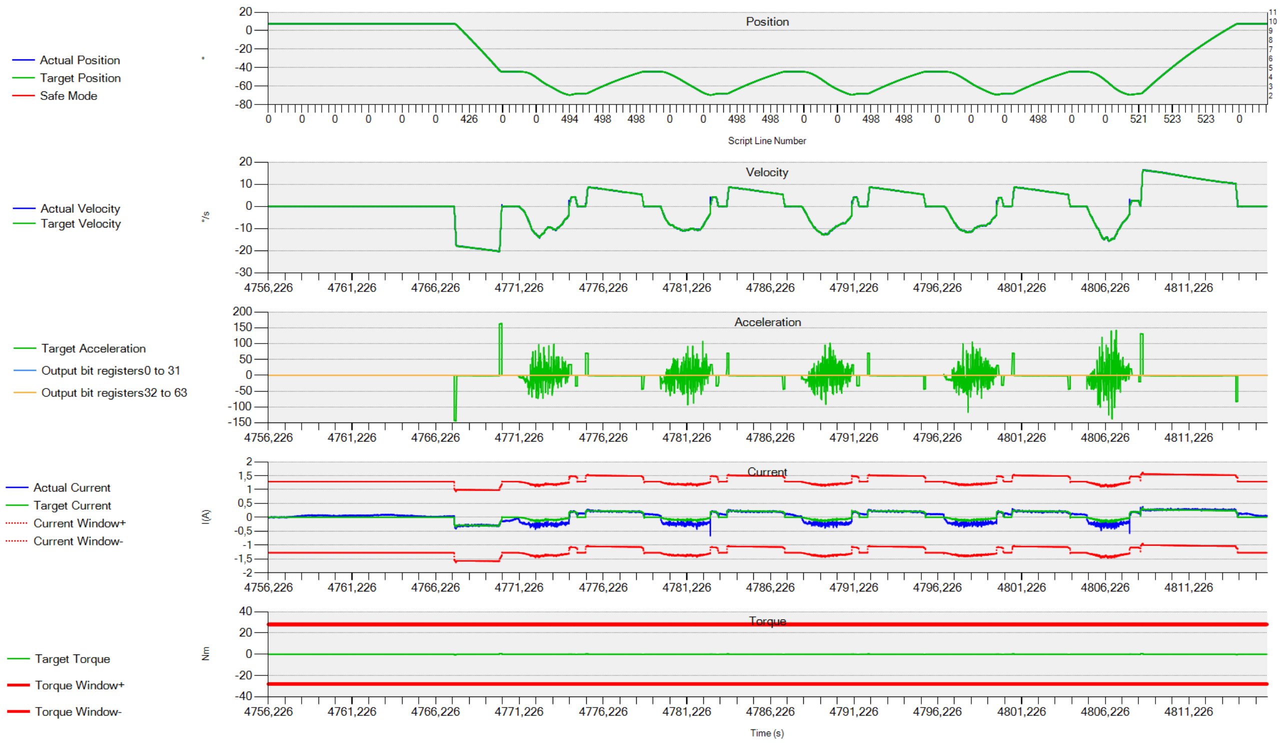


Figure 156 - Healthy subject 18. Right Side. M-Modality.

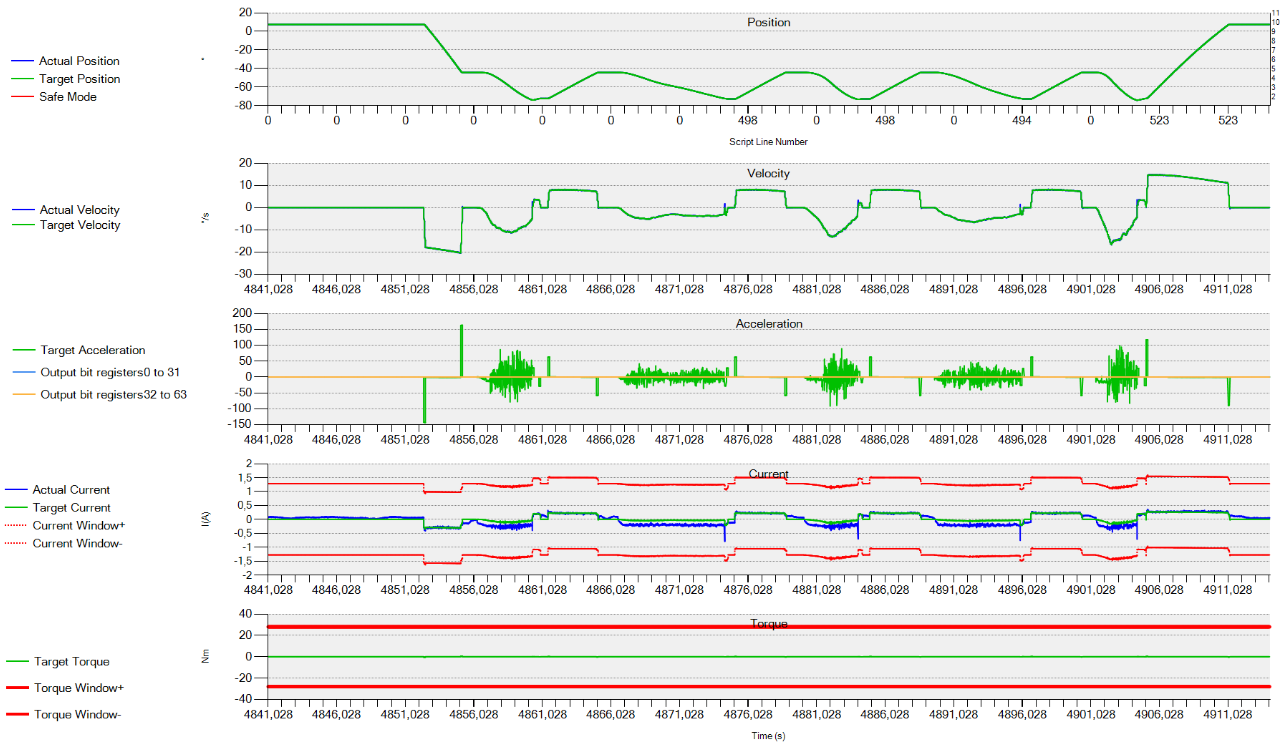


Figure 157 - Healthy subject 18. Right Side. D-Modality.

HEALTHY SUBJECT 19

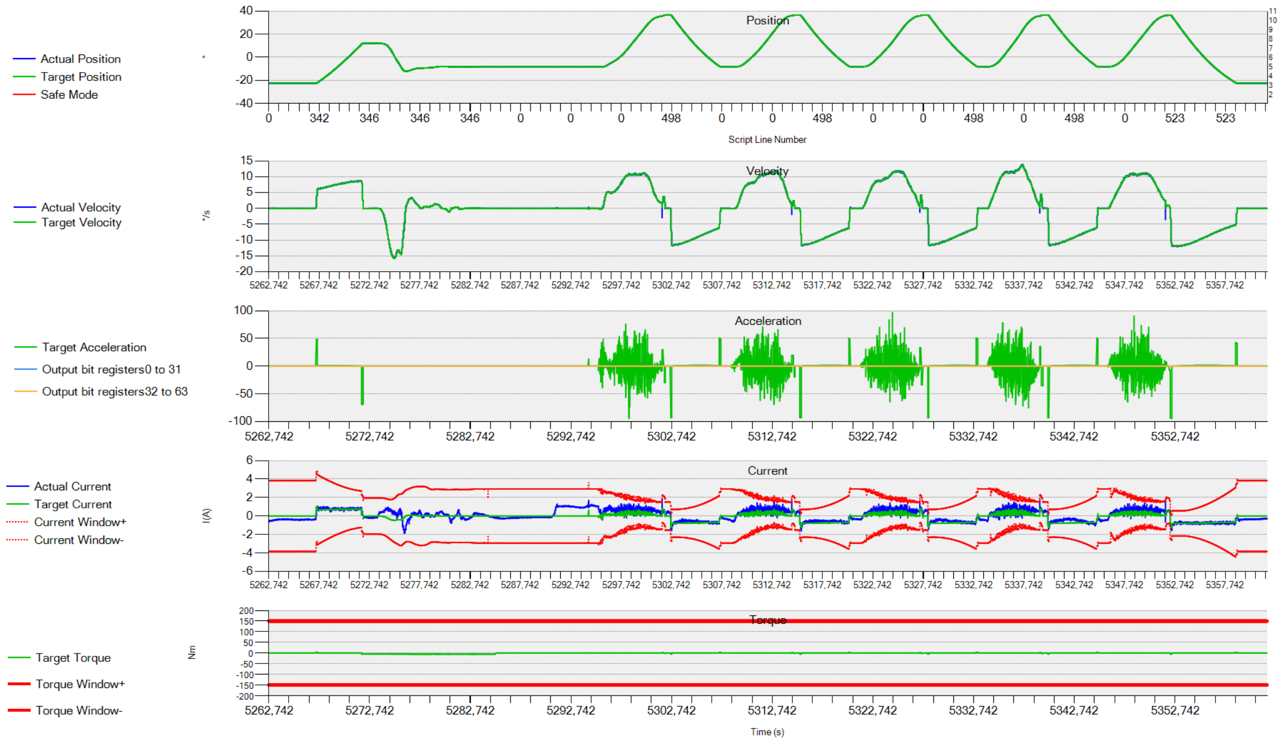


Figure 158 - Healthy subject 19. Left Side. F-Modality

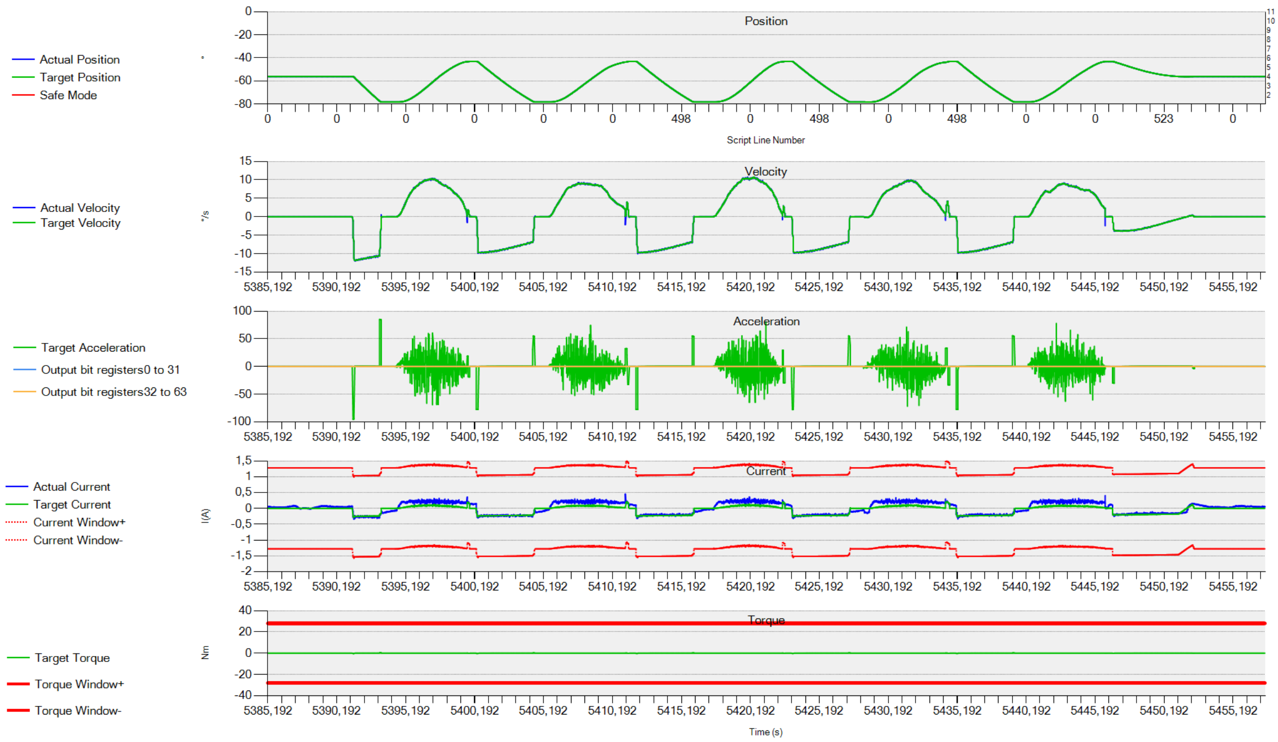


Figure 159 - Healthy subject 19. Left Side. M-Modality.

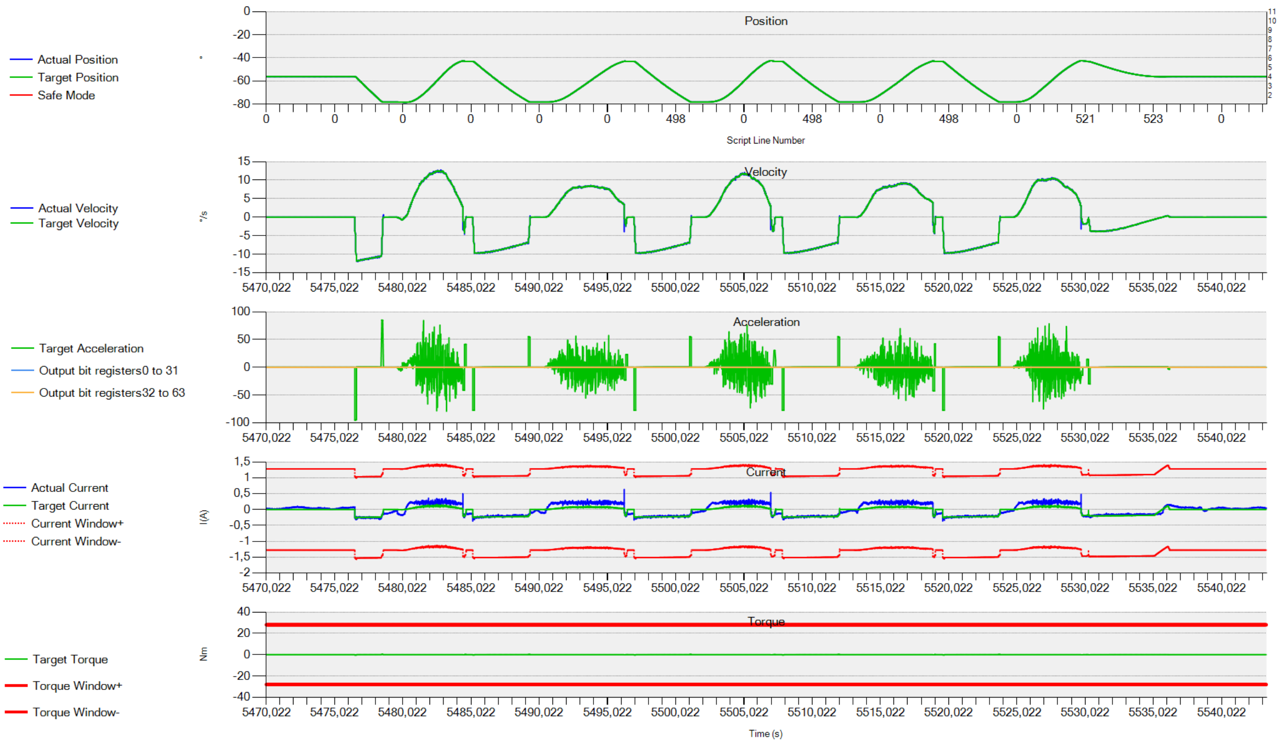


Figure 160- Healthy subject 19. Left Side. D-Modality.

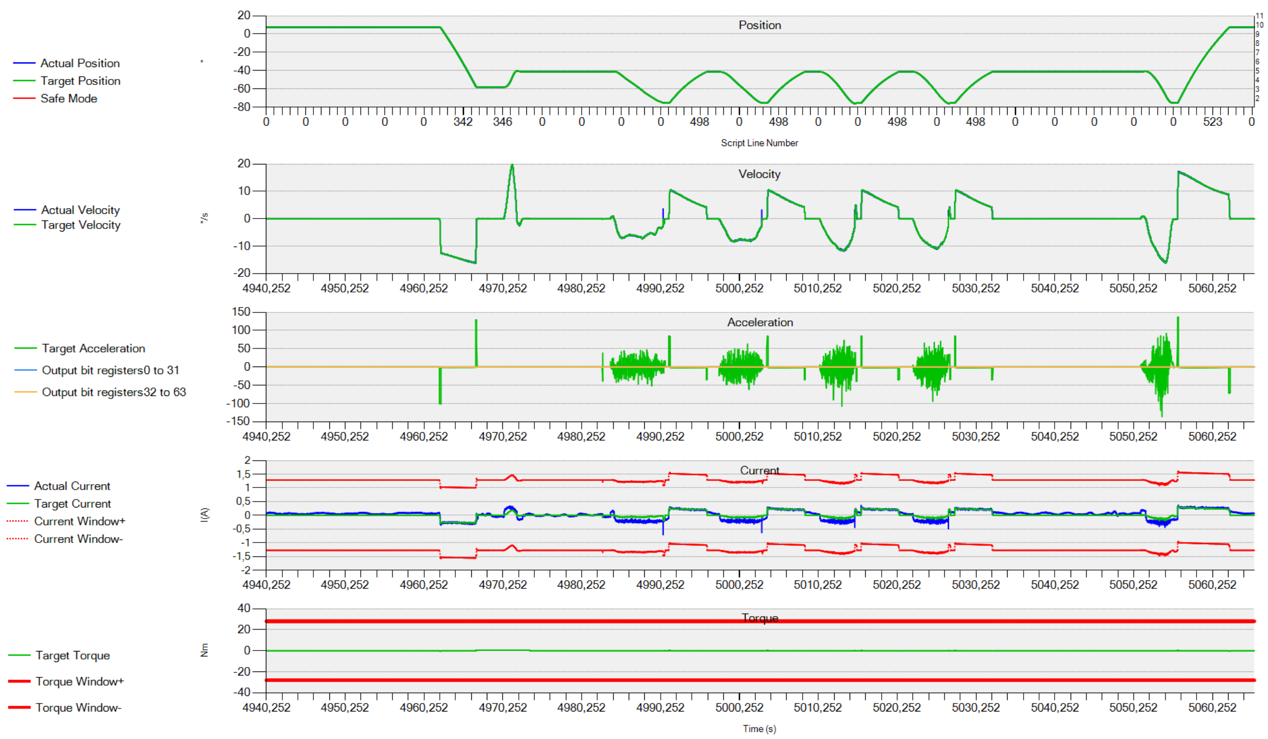


Figure 161 - Healthy subject 19. Right Side. F-Modality.

Figure 162 - Healthy subject 19. Right Side. M-Modality.

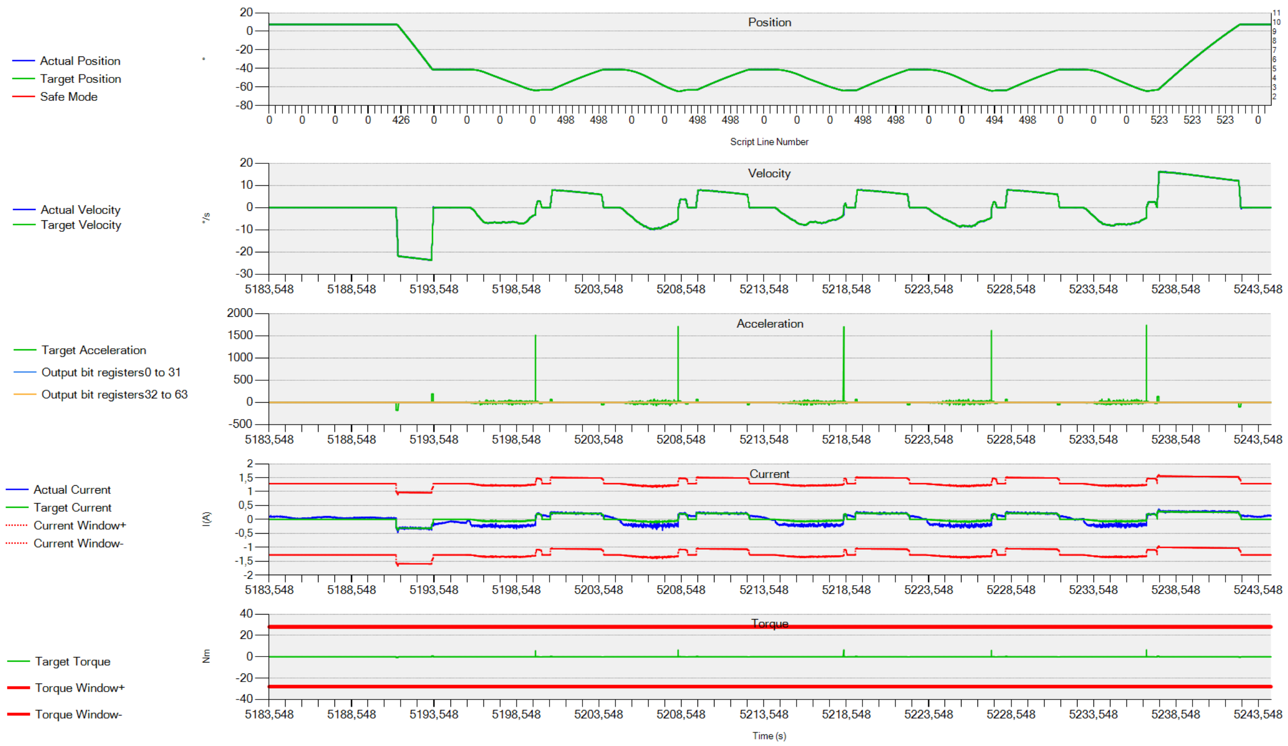


Figure 163 - Healthy subject 19. Right Side. D-Modality.

Part D - Matlab Code for Force Analysis

```
%% post process rehabilitation

clear
clc
close all



for paz = [2];
%for braccio = ['s'];
    for braccio = ['s','d'];
%for mod = ['d'];
    for mod = ['f','m','d'];

clear q qd Pf

filename = ['P',num2str(paz),'_',braccio,'_',mod,'.csv'];

dataRaw = readtable(filename);

% create timestamp
time = dataRaw.timestamp;
time = time-time(1);

% joints
q(:,1) = dataRaw.actual_q_0;
q(:,2) = dataRaw.actual_q_1;
q(:,3) = dataRaw.actual_q_2;
q(:,4) = dataRaw.actual_q_3;
q(:,5) = dataRaw.actual_q_4;
q(:,6) = dataRaw.actual_q_5;

qd(:,1) = dataRaw.actual_qd_0;
qd(:,2) = dataRaw.actual_qd_1;
qd(:,3) = dataRaw.actual_qd_2;
qd(:,4) = dataRaw.actual_qd_3;
qd(:,5) = dataRaw.actual_qd_4;
qd(:,6) = dataRaw.actual_qd_5;

% cartesian
for cont=1:size(q,1)
    [P,~,~] = UR5e_FK(q(cont,:));
    TCP(cont,:) = P';
end

% sensor force
F_TCP = [dataRaw.actual_TCP_force_0 dataRaw.actual_TCP_force_1
dataRaw.actual_TCP_force_2];

% phases
trig = dataRaw.actual_digital_output_bits;
trig(find(trig==0)) = 4;
trig = (trig - min(trig))./(max(trig) - min(trig));

trig_up = find(diff(trig)==1)+1;

bips = numel(trig_up);
if mod=='f'
    n_reps = (bips-2-3)/3;

    set_up = [1 trig_up(1)];
    calib_f = [trig_up(1) trig_up(2)];
    f_max = [trig_up(2) trig_up(3)];

    reps = [];
    backs = [];
    for cont=1:n_reps
        reps = [reps; trig_up(3*cont) trig_up(3*cont+2)];
        backs = [backs; trig_up(3*cont+2) trig_up(3*cont+3)];
    end

    % starting point
    Pi = mean(TCP(calib_f(1) : calib_f(2),:))';

    F0 = F_TCP([calib_f(1):calib_f(2)],:);

    timeF0 = time([calib_f(1):calib_f(2)]);
    index = find(timeF0-timeF0(end)>-2);

    F0 = norm(mean(F0(index,:)));

    eval(['F0_', braccio,'=',num2str(F0),';']);

    Fmax = F_TCP([f_max(1):f_max(2)],:);
    Fmax = max(sqrt(sum(Fmax'.*Fmax')));

    eval(['Fmax_', braccio,'=',num2str(Fmax),';']);
else
    n_reps = (bips-3)/3;

    set_up = [1 trig_up(1)];

    reps = [];
    backs = [];
    for cont=1:n_reps
        reps = [reps; trig_up(3*cont-2) trig_up(3*cont)];
        backs = [backs; trig_up(3*cont) trig_up(3*cont+1)];
    end
end

% target point
for cont = 1:n_reps-1
    t_back = time(backs(cont,1):backs(cont,2));
    t_back = t_back-t_back(1);

    TCP_back = TCP(backs(cont,1):backs(cont,2),:);

    index = find(t_back>=0.3 & t_back<=0.6);
    Pf(cont,:) = mean(TCP_back(index,:));
end
Pf = mean(Pf)';

% line feature
e3 = (Pi-Pf)./(sqrt((Pi-Pf)'*(Pi-Pf)));
```

```
e31 = e3(1);
e32 = e3(2);
e33 = e3(3);

e11 = -e32*(1/(e31^2 + e32^2))^(1/2);
% e32*(1/(e31^2 + e32^2))^(1/2)
e12 = e31*(1/(e31^2 + e32^2))^(1/2);
% -e31*(1/(e31^2 + e32^2))^(1/2)
e1 = [e11 e12 0]';

e2 = cross(e3,e1);

Rl = [e1 e2 e3];

Tl = [Rl Pf; 0 0 0 1];

% elastic force
d_max = sqrt((Pi-Pf)'*(Pi-Pf));
r_max = d_max/5;
k_max = 2000;
perc = [0.5 0 -0.5];

% - 2*k*r^3 + 3*k*r^2

for cont=1:size(q,1)
   % calcolo forza radiale
   TCPr(cont,:) = (eye(3,4)*(Tl\[TCP(cont,:) 1]'))';

   d = abs(TCPr(cont,3))/d_max;
   r_d = d*r_max/d_max;

   r(cont) = sqrt(TCPr(cont,1:2)*TCPr(cont,1:2)');

   if r(cont)/r_d>1
     k(cont) = k_max;
   else
     k(cont) = -2*k_max*(r(cont)/r_d)^3 + 3*k_max*(r(cont)/r_d)^2;
   end

   Frl(cont,1:2) = k(cont)*TCPr(cont,1:2);
   Fr(cont,1:3) = Rl*[Frl(cont,1:2) 0]';

   % calcolo forza longitudinale

   Fl(cont,:) = Rl*F_TCP(cont,:)';

   if mod=='f'
     Frob(cont,1) = Fl(cont,3)*perc(1);
   elseif mod=='m'
     Frob(cont,1) = Fl(cont,3)*perc(2);
   else
     Frob(cont,1) = Fl(cont,3)*perc(3);
   end



end

% PLOT
figure

if braccio=='s'
sgtitle(['\fontsize{12}{0}\selectfont Paziente ',num2str(paz),', braccio
sinistro, ',mat2str(n_reps),' ripetizioni, mod. ',mod,'.', char(10), ...
     '\fontsize{9}{0}\selectfont $F_0=',num2str(fix(F0_s*10)/10),'$ N,
$F_{max}=',num2str(fix(Fmax_s*10)/10),'$ N'], ...
     'interpreter','latex');
elseif braccio=='d'
sgtitle(['\fontsize{12}{0}\selectfont Paziente ',num2str(paz),', braccio destro,
',mat2str(n_reps),' ripetizioni, mod. ',mod,'.', char(10), ...
     '\fontsize{9}{0}\selectfont $F_0=',num2str(fix(F0_d*10)/10),'$ N,
$F_{max}=',num2str(fix(Fmax_d*10)/10),'$ N'], ...
     'interpreter','latex');
end

% traiettoria
subplot(7,1,1:4)
title('Traiettorie eseguite','interpreter','latex')
hold on
axis equal
for cont=1:size(reps,1)

plot3(TCP(reps(cont,1):reps(cont,2),1),TCP(reps(cont,1):reps(cont,2),2),T
CP(reps(cont,1):reps(cont,2),3));
end
grid
box
set(gca,'view',[105 15])
line([1 0 0]*[Pi Pf],[0 1 0]*[Pi Pf],[0 0 1]*[Pi Pf],'linestyle','--
','marker','.','markersize',10,'color','black')

xlabel('$x$ (m)','interpreter','latex')
ylabel('$y$ (m)','interpreter','latex')
zlabel('$z$ (m)','interpreter','latex')

xlim([0.4 0.8])
ylim([-0.25 0.25])
zlim([-0.05 0.25])

r = linspace(0,d_max) ;
th = linspace(0,2*pi) ;
[R,T] = meshgrid(r,th) ;
X = r_max/d_max*R.*cos(T) ;
Y = r_max/d_max*R.*sin(T) ;
Z = R;
for cont=1:numel(R)
   CN = Tl*[X(cont) Y(cont) Z(cont) 1]';
   X(cont) = CN(1);
   Y(cont) = CN(2);
   Z(cont) = CN(3);
end
surf(X,Y,Z,'edgealpha',0,'FaceColor',0.5*ones(1,3),'FaceAlpha',0.1)


% forza radiale
subplot(7,1,5)
title('Modulus of radial force','interpreter','latex')
hold on
for cont=1:size(reps,1)

   Tn = (time(reps(cont,1):reps(cont,2))-
time(reps(cont,1)))./(time(reps(cont,2))-time(reps(cont,1)));

plot(Tn,sqrt(sum((Frl(reps(cont,1):reps(cont,2),1:2).*Frl(reps(cont,1):reps(
cont,2),1:2))')));
```

```
end
grid
box

xlabel('$t_n$','interpreter','latex')
ylabel('$F_r$ (N)','interpreter','latex')


% forza assiale
subplot(7,1,6)
title('Modulus of the measured axial force','interpreter','latex')
hold on
for cont=1:size(reps,1)

    Tn = (time(reps(cont,1):reps(cont,2))-
time(reps(cont,1)))./(time(reps(cont,2))-time(reps(cont,1)));
    plot(Tn,Fl(reps(cont,1):reps(cont,2),3));

end
grid
box

xlabel('$t_n$','interpreter','latex')
ylabel('$F_l$ (N)','interpreter','latex')

% forza robot
subplot(7,1,7)
title('Modulus of the force exerted by the robot','interpreter','latex')
hold on
for cont=1:size(reps,1)

    Tn = (time(reps(cont,1):reps(cont,2))-
time(reps(cont,1)))./(time(reps(cont,2))-time(reps(cont,1)));
    plot(Tn,Frob(reps(cont,1):reps(cont,2),1));

end
grid
box

xlabel('$t_n$','interpreter','latex')
ylabel('$F_{rob}$ (N)','interpreter','latex')



end
end
end
```

References

[1] S. Molinelli, “"Anatomia funzionale e patologie più frequenti della spalla",” 2001. [Online]. Available: www.fisiobrain.com.

[2] N. Wuelker, M. Korell and K. Thren, “Dynamic glenohumeral joint stability,” *Journal of shoulder and elbow surgery,* pp. vol. 7, n°1, pag. 43-52, 1998.

[3] G. Legnani and G. Palmieri, “Fondamenti di meccanica e biomeccanica del movimento,” Città Studi Edizioni, 2016.

[4] P. Langhorne, F. Coupar and A. Pollock, “Motor recovery after stroke: a systematic review.,” *The Lancet,* 2009.

[5] A. Sunderland, D. J. Tinson, E. L. Bradley, D. Fletcher, R. L. Hewer and D. T. Wade, “Enchanced physical theraphy improves recovery of arm function after stroke: a randomized control trial.,” *Journal of Neurology, Neurosurgery and Phychiatry,* 1992.

[6] S. Hatem, G. Saussez, M. D. Faille, V. Prist, X. Zhang, D. Dispa and Y. Bleyenheuft, “Rehabilitation of Motor Function after Stroke: A Multiple Systematic Review Focused on Techniques to Stimulate Upper Extremity Recovery.,” *Frontiers in Human Neuroscience,* p. 442, 16 September 2016.

[7] H. Krebs, N. Hogan, M. Aisen and B. Volpe, “Robot-aided neurorehabilitation,” *IEEE Transactions on Rehabilitation Engineering,* vol. 6, pp. 75-78, 1998.

[8] D. J. Reinkensmeyer, L. E. Kahn, M. Averbuch, A. McKenna-Cole, B. D. Schmit and W. Z. Rymer, “Understanding and treating arm movement impairment after chronic brain injury: Progress with the ARM guide,” *Journal of Rehabilitation Research and Development,* vol. 37, pp. 653-62, 2000.

[9] P. Lump, C. G. Burgar, P. Shor, M. Majmundar and M. Van Der Loos, “Robot-assisted movement training compared with conventional therapy techniques for the rehabilitation of upper-limb motor function after stroke,” *Archives of Physical Medicine and Rehabilitation,* vol. 83, pp. 952-959, 2002.

[10] U. Robots, Script Manual - The URScript Programming Language.

[11] A/S, Universal Robots, *UR Log Viewer Manual: e-Series and CB-Series,* 2009–2020, Original Instructions (EN), Documentation version: 1.2.