RAM ROBOTICS AND MECHATRONICS

Joint level modelling, characterisation and torque control of the SHERPA robotic arm

J. (Jeroen) Jager

MSc Report

Committee:
Prof.dr.ir. S. Stramigioli
Dr. R. Carloni
E. Barrett, MSc
Dr.ir. R.G.K.M. Aarts

August 2017

042RAM2017
Robotics and Mechatronics
EE-Math-CS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE.

MIRA CTIT
BIOMEDICAL TECHNOLOGY AND TECHNICAL MEDICINE

Summary

The SHERPA project is a European project whose goal is to develop a mixed ground and aerial robotic platform to support search and rescue activities in a real-world environment. The following actors compose the basic 'SHERPA team': a human rescuer, small scale rotary-wing Unmanned Aerial Vehicles (UAVs), a ground rover and long endurance, high-altitude, high-payload aerial vehicles. In order to improve the autonomous capabilities of the robotic platform, a multi-functional robotic arm is installed on the rover(Marconi et al., 2012) (Barrett et al., 2017).

The University of Twente research team has developed a 7 active Degrees of Freedom (DOF) robotic arm for the ground rover. The arm is equipped with two variable stiffness actuators. The VSA enables the robotic arm to become compliant and adapt its dynamic behaviour to different tasks. High level control software is implemented in ROS running on a Intel NUC running Ubuntu, to be able to control the robotic arm in the workspace.

In case of the SHERPA project the dynamic environment is unknown and interaction tasks with humans or its dynamic environments is required. A particular popular control method to be able to interact with the environment is the impedance control formulation. In order to obtain impedance control on the SHERPA robotic arm, joint level torque control is required. This master thesis focuses on identification, modelling of the joint characteristics and torque control by means of the arm's variable stiffness actuators.

Torque control on the SHERPA robotic arm is achieved by regulating the deflection of the compliant joints. The output position of the robotic arm is fixed, such that during torque control the robotic arm is in contact with the environment. For each joint a dynamic model in bond-graphs is created and implemented in 20-sim. Simulations are performed to determine the behaviour and characteristics of each joint. The joints are characterised in a test set-up and curve fitting is applied to obtain the torque-deflection characteristics. The acquired torque-deflection characteristics are implemented in the high-level software. A control loop with a PI controller is written to perform torque control. Joint controllers convert each desired joint position into motor positions, which are sent to the ELMO motor drives. These will perform local position control to achieve the given set-point.

Overall the torque control for the variable stiffness joints provides a good tracking of the reference torque and is validated with experiments. With this approach joint level torque control for the variable stiffness joints is accomplished and the objectives for this master thesis are satisfied.

Contents

1 Introduction

1.1 The SHERPA project

The SHERPA project is a European project whose goal is to develop a mixed ground and aerial robotic platform to support search and rescue activities in a real-world environment. The following actors compose the basic 'SHERPA team' (Marconi et al., 2012) (Barrett et al., 2017):

- A human rescuer, who is an expert of the specific rescuing mission or surveillance activity (such as a mountain guide or a forest guard), is naturally a member of the rescue team.
- Small scale rotary-wing Unmanned Aerial Vehicles (UAVs), equipped with small cameras and other sensors/receivers, are used to support the rescue and surveillance mission.
- A ground rover serves as a transportation module for the rescuer, hardware station with computational and communication capabilities and as a recovery and recharging module for the small-scale unmanned aerial vehicles introduced above. In order to improve the autonomous capabilities of the robotic platform, a multi-functional robotic arm (hereafter named arm) is also installed on the rover. The main task of the arm is to autonomously recover and deploy the small scale UAVs.
- Long endurance, high-altitude and high-payload aerial vehicles, with complementary features with respect to the small-scale UAVs introduced before. They are used for constructing a 3D map of the rescuing area, as communication hub between the platforms in presence of critical terrain morphologies and for patrolling large areas.

The University of Twente research team has developed a 7 Degrees of Freedom (DOF) robotic arm for the ground rover. The arm is equipped with two variable stiffness actuators, hereafter named VSA. The VSA enables the arm to become compliant and adapt its dynamic behaviour to different tasks. The compliance increases the level of safety during human-robot interaction or with its environment (Groothuis et al., 2015). Furthermore increased robustness against external impacts together with high dynamic performance and energy efficiency due to the energy storing capabilities can be achieved with VSA robots (Petit et al., 2015a).

1.2 Problem statement

In many robotic tasks where paths and relationships are specified in a geometric manner position control is sufficient. Such an approach requires a structured and predictable environment, where interaction is known in advance. This prerequisite is often not given for physical interaction tasks with humans or in dynamic environments (Petit, 2014). In case of the SHERPA project the dynamic environment is unknown and interaction task with humans or its dynamic environments is required. A particular popular control method to be able to interact with the environment is the impedance control formulation. This approach provides an intuitive interpretation, resulting in that the controlled robot behaves like a mass-spring-damper system. The torque control technology shows great success, both in the research community and in industrial applications (Petit, 2014). However challenges are the non-linear robot dynamics and the non-linear VSA characteristics, which shows low intrinsic damping. (Petit et al., 2015a). In order to obtain impedance control on the SHERPA arm, joint level torque control is fundamental. This master thesis focuses on identification, modelling of the joint characteristics and torque control by means of the arm's variable stiffness actuators. The objectives for this master thesis are:

- Formulating a dynamic model of the arm's joints.

- Characterisation of the torque-deflection properties of the joints.
- Design of a state observer based on the arm joints.
- Design of a low-level torque controller on the (variable stiffness) joints.

1.3 Report outline

The remainder of this thesis is organized as follows. A detailed hardware description of the arm is provided in chapter 2. After the hardware description each joint is modelled and characterised. From these characterised joint's, a state observer and joint level torque controller is proposed in chapter 3. The resulted joint level torque controllers are validated in experiments, these results are presented in chapter 4. In chapter 5 conclusions are drawn and recommendations for future work are proposed.

2 The SHERPA robotic arm

In this chapter background information about the robotic arm is presented and the approach for modelling, characterisation and the control architecture is presented and discussed.

2.1 Robotic arm

The arm is a 7 active degree of freedom (DOF) robotic arm, of which an overview is given in figure 2.1. Three DOF (q0 till q2) are located in the shoulder, the second and third DOF (q2, q3) are actuated by two differential coupled actuators. The shoulder contains one VSA mechanism dedicated to the second DOF (q2). The elbow (q4) contains one DOF. The elbow consist of two axes, which are attached by a intermediate link. The elbow is driven by a single actuator. The last 3 DOF are located in the wrist (q4 till q7), which are driven by 3 actuators. A single actuator drives 1 DOF (q5). The other two actuators are coupled using a bevel differential and drive 2 DOF (q6 and q7). Additionally the wrist contains a VSA dedicated for q5 and q6. On top of the wrist a gripper and camera are mounted to be able to pick up a drone.

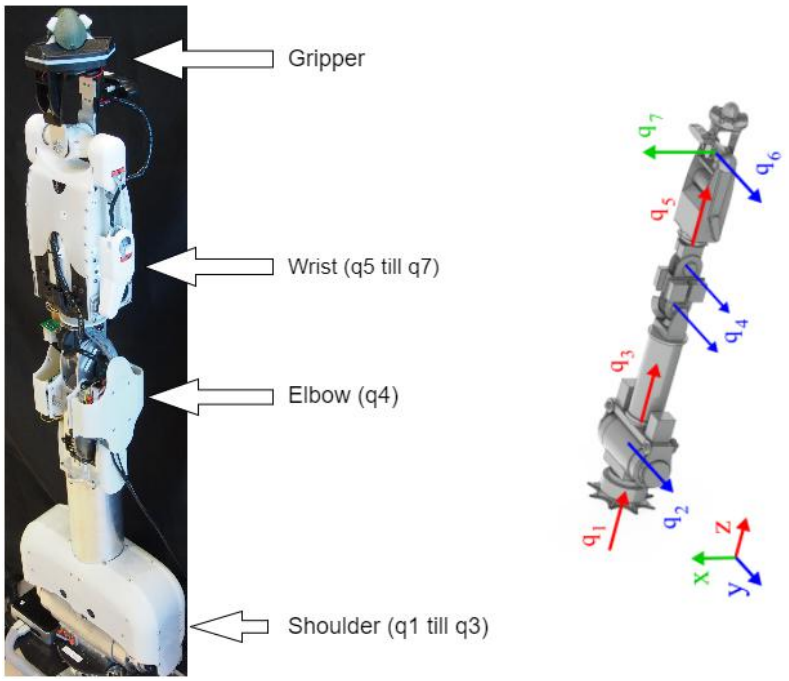


Figure 2.1: Overview of the 7 DOF arm (Barrett et al., 2017).

2.2 Electronics

The arm currently contains 9 ELMO Whistle miniature digital servo drives (Elmo, 2017) that locally perform position control on each actuator. Each actuator is equipped with an incremental encoder, which provides the position feedback for the ELMO. Besides incremental encoders the arm is equipped with 13, 14-bit absolute magnetic encoders. All high level control for the arm is performed on a Intel NUC running Ubuntu 14.04. The communication between the high level control and the ELMO whistles is realized through a CAN bus. The absolute encoders communicate using a SPI bus. The gripper is connected with the high level control using a serial interface and the camera with a USB connection. An overview of the hardware and various communication layers are presented in figure 2.2.

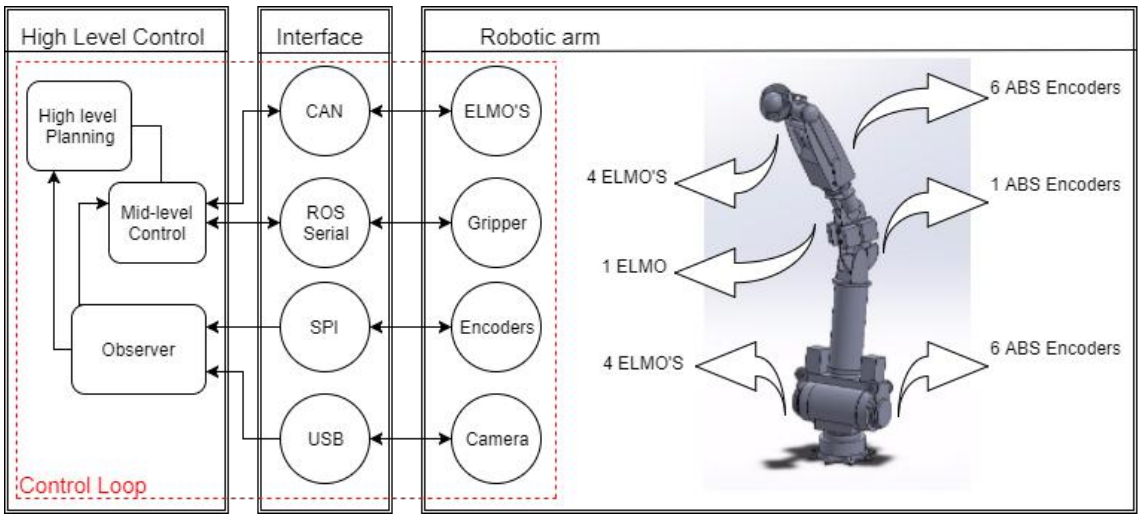


Figure 2.2: Hardware components and the various communications layers on the arm.

2.3 Software architecture

High level control software is implemented in ROS running on the NUC, to be able to control the arm in the workspace. To avoid self collision and collision with objects, various state machines have been implemented together with Moveit!. MoveIt! is a state of the art software for manipulation, incorporating the latest advances in motion planning, manipulation, 3D perception, kinematics, control and navigation. It provides an easy-to-use platform for developing advanced robotics applications, evaluating new robot designs and building integrated robotics products for industrial, research and development, commercial and other domains (MoveIt!, 2017).

The observer shown in figure 2.2, provides the high level planner with all the necessary information about the current state of each joint, for example the position. The high level planner will construct a collision free motion path using the Moveit! package, taking into consideration joint limits such as maximal accelerations. This path is sent to the Mid-level controller, which will execute the path and converts each desired joint position into motor set-points. These set-points are sent to ELMO's by means of the CAN bus. These ELMO's will perform local position control to achieve the given set-point. The complete high level control on the NUC is executed with a loop-rate of 40 Hz. In addition the communication structure introduces latency in the high level control. The latency of the SPI bus is approximately 15 milliseconds and for the CAN bus the latency is approximately 30 milliseconds.

2.4 Variable stiffness actuators

The arm contains two VSA's, one for the shoulder (q2) and one for the wrist (q5 and q6). These VSA's are based on a variable transmission. An overview of the internal components of the VSA and its working principle is depicted in figure 2.3. The mechanical design of both VSA's are different, but it's working principle is the same.

A lever-arm based stiffness adjustment mechanism realizes a variable transmission between the output and the internal springs by changing the lever arm ratio between the two, which is achieved by moving the pivot point along the lever. The output stiffness is described with the equation 2.6. However these equations are derived under the approximations for small output deflections, but still offer a good representation of the mechanisms properties and are a useful tool for analysis (Barrett et al., 2016).

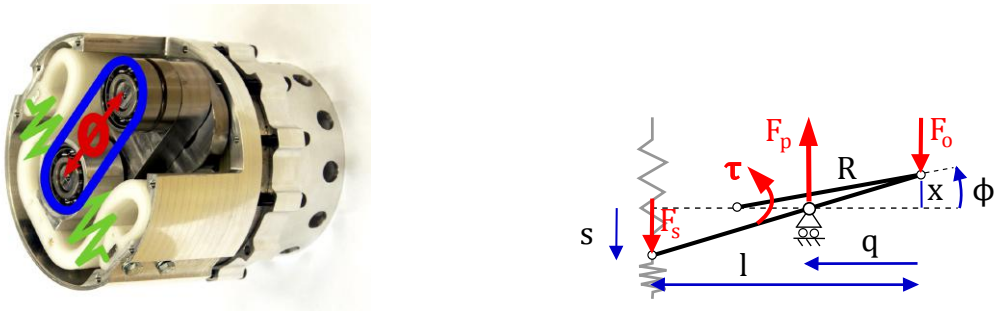


Figure 2.3: Overview of the VSA in the shoulder and the working principle of the VSA. The lever arm (blue) connects the internal polymer spring (green) to the output, through a variable transmission that is defined by the position of the movable pivot point (red) of the lever (Barrett et al., 2016).

A crank with length R transforms the linear force F_o and deflection x of the lever into the rotational torque τ and deflection ϕ. A deflection $x = \sin(\phi)R$ of the lever at the connection with the output causes an elongation s of the spring (Barrett et al., 2016).

$$s = \frac{l-q}{q}x = \frac{l-q}{q}\sin(\phi)R \tag{2.1}$$

In this equation l is the lever length and q the position of the pivot point of the lever. The internal forces on the lever at the connection points, output and pivot are described as follows. In these equations k_s is the elastic constant of the internal springs.

$$F_s = k_s s \tag{2.2}$$

$$F_o = \frac{l-q}{q}F_s \tag{2.3}$$

$$F_p = F_s + F_o \tag{2.4}$$

It follows that $\tau = F_o R$. Substituting all variables in the equations results in the output torque and taking the derivative with respect to ϕ results in the output stiffness.

$$\tau = \frac{l-q}{q}Rk_s s = (\frac{l-q}{q}R)^2 k_s \phi \tag{2.5}$$

$$K = \frac{d\tau}{d\phi} = (\frac{l-q}{q}R)^2 k_s \tag{2.6}$$

The output stiffness can be adjusted by varying q and scales linearly with k_s. The position of the pivot point, q is changed with an actuator (Barrett et al., 2016).

2.5 Torque control methods

Robots with elastic joints arouse an increasing interest in the robotics community. Especially, variable stiffness robots promise to be beneficial regarding robustness and task adaptability. However, the control of variable stiffness (multi-joint) robots is not straight forward and challenging due to their highly non-linear behaviour (Petit et al., 2015a). In general torque control can be achieved by regulating the deflection of the compliant element or with current control such that the output torque can be controlled with high accuracy. A crucial factor is the joint torque estimation capability of VSAs (Petit et al., 2015b). The VSA joint characteristics with a

non-linear and variable spring torque function are described as $\tau = f(\varphi, \sigma)$. Where τ is the joint torque, φ is the joint deflection and σ is the stiffness variation parameter.

Possible control solutions for VSA joints are based on cascaded structures, integrator backstepping, feedback linearisation and integral manifold control (Keppler et al., 2016). Another approach is command filtering backstepping control. This method provides tracking control while still being able to cope with sensor quantization and noise, actuator limitations, signal delays, and unmodeled dynamics. The control law is based on a PD+ controller. Disadvantageous are the dependence on a precise dynamic model. Model errors and unmodeled dynamic effects such as actuator friction lead to non-vanishing control errors. Furthermore the choice of correct filter gains is critical as a balance between noise suppression and introduced delay has to be found (Petit et al., 2015a).

The hardware and software architecture of the arm constrain possible control solutions. The current and proposed control solutions are based on one or multiple of the following hardware and software requirements:

- Control electronics allowing local, collocated current/torque control on the motor.
- Precise dynamic model.
- Full state feedback of motor and output positions, velocities and torque.
- Higher order derivatives of the state feedback.
- High frequency control loop.

The software and hardware architecture on the arm does not meet these requirements. The ELMO servo drives can only perform position control and show non-linear behaviour that is challenging to model, resulting in a inaccurate dynamic model. Furthermore the communication structure introduces latency in the state feedback and control action.

It is therefore chosen to control the deflection of each joint at low frequencies. By controlling the joint deflection, the joint output torque is controlled. For each joint a dynamic model is created and simulations are performed to determine the behaviour and characteristics of each joint. Since the deflection is only controlled for low frequencies, bode plots no longer provide insight in the joints behaviour. Each joint is then characterised in a test set-up to obtain the torque-deflection characteristics. The deflection is controlled in the high-level controller running on the NUC and will provide the ELMOs with position set-points. The ELMOs ensure that the actuator reaches its desired position. The output position of the arm is fixed, such that during torque control the arm is in contact with the environment. Using this approach it is possible to control the output torque of each compliant joint. In the next chapter the joints are modelled and characterised.

3 Modelling and characterisation

3.1 Introduction

The joints of the arm are separated into rigid, elastic and variable stiffness joints, of which an overview is given in figure 3.1. The arm contains 2 rigid joints, which are located in the shoulder (q1 and q3). These joints are a combination of a actuator with a harmonic drive (CSD-32-100-2UH) gearbox with a stiffness of $8.3 * 10^4 [\frac{Nm}{RAD}]$, such that the joint can be seen as rigid in comparison with the other joints (Harmonic, 2017). Therefore it is chosen not to model and characterise these joints. The 2 elastic joint are the elbow (q4) and the wrist (q5). The 3 variable stiffness joints are the shoulder joint (q2) and the wrist joints (q6 and q7).

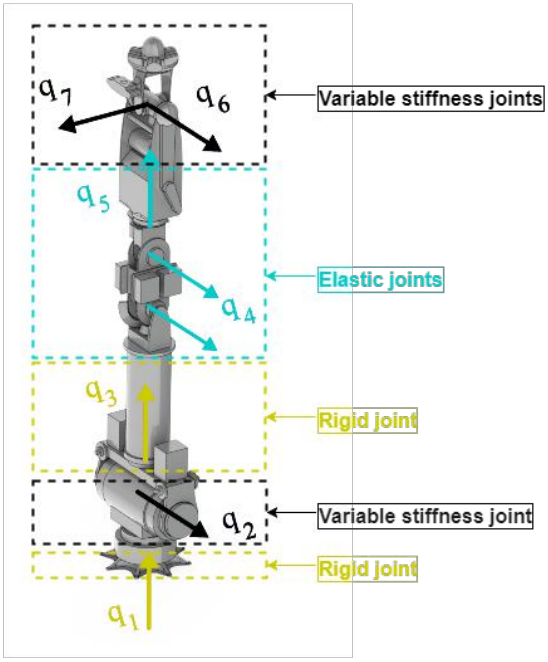


Figure 3.1: Different types of joints of the arm.

First the motor model and elmo controller are modelled since these are similar in all joints. The elastic joints are then modelled and characterised. After the elastic joints the variable stiffness joints are modelled and characterised.

3.2 ELMO motor drives

The motor model and ELMO controllers are similar in all joints. Documentation of the ELMO servo controllers provides a control scheme in which the position, velocity and current control are cascaded (ELMO, 2017). This provided control scheme is depicted in figure 3.2. Within these control loops various filter and additional feed-forward terms are present. The exact gains and filters are not stated in the documentation. An attempt is made to duplicate the exact behaviour and control loop of the ELMO using the provided control scheme in with bond graphs in 20-sim. Bond graphs are a network-like description of physical systems in terms of ideal physical processes. With the bond graph method, the system characteristics are modelled into a set of separate elements. Each element describes an idealized physical process (20sim, 2017). However without the knowledge of the exact gains and filtering, it was not possible to duplicate the behaviour of the ELMO with the provided control scheme.

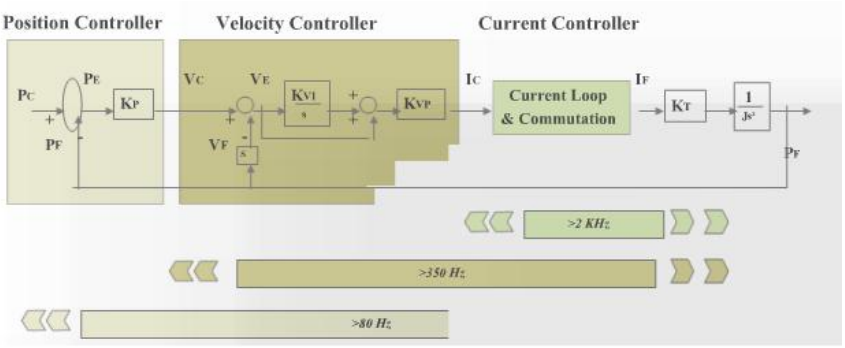
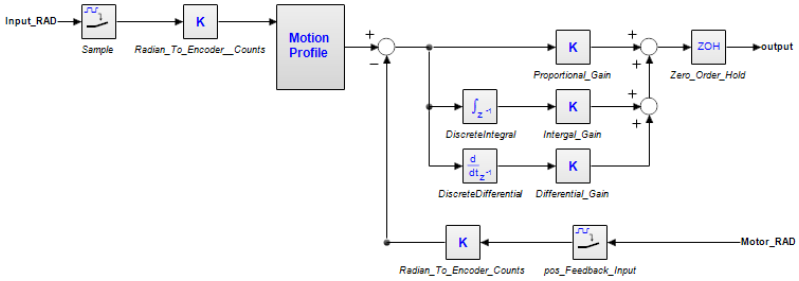
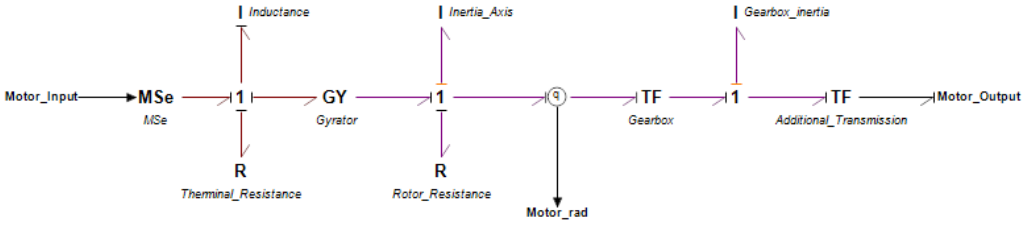


Figure 3.2: Control loop of the ELMO servo drives (ELMO, 2017).

Through practical experience with the arm and ELMO servo drives, it was concluded that the ELMO constructs a motion profile based on the maximal acceleration and velocity. This profile was followed by the ELMO controller in all encountered situations. It has been chosen to duplicate this motion profile combined with a controller and motor model in 20-sim. The desired motor position is controlled with a PID controller with encoder feedback from the motor. The gains where tuned in such a way that each desired motor position is achieved. The ELMO Controller model is presented in 3.3 a and the motor model in figure 3.3 b.



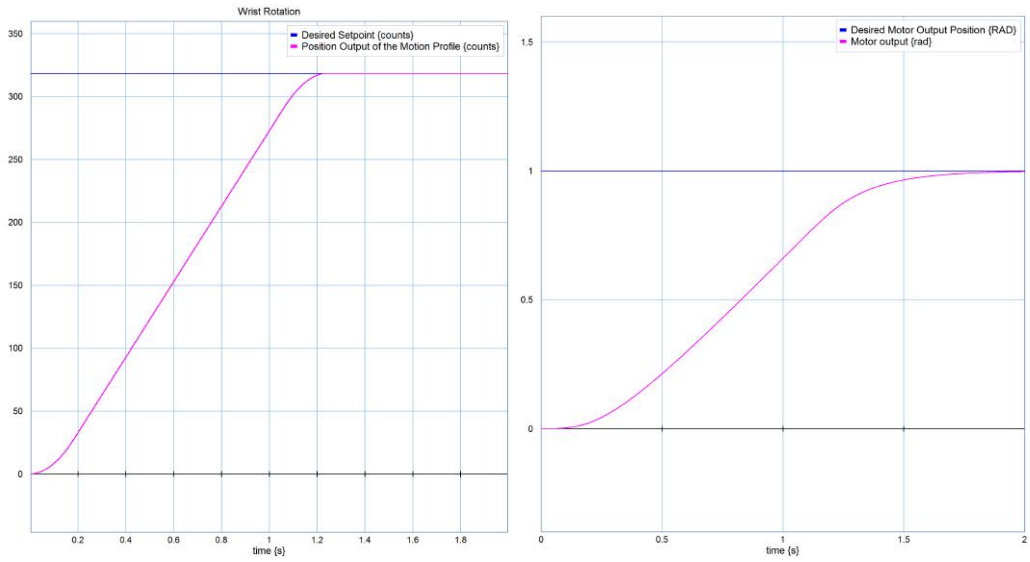
(a) 20-sim model of the ELMO controller.



(b) 20-sim model of the motor.

Figure 3.3: Overview of the 20-sim model of the ELMO controller and motor model.

In figure 3.4 the result of the duplicated ELMO servo drive is depicted. In this simulation a desired motor output position of 1 radian is set. The maximal acceleration is 5 $\frac{counts}{s^2}$, the maximal velocity is set to 300 $\frac{counts}{s}$. The motion profile accelerates until the maximal velocity is reached and maintains this velocity. The starting point of deceleration is calculated based on the current velocity and position, taken in consideration the maximal deceleration. This prevents a overshoot of the desired set-point. The motor output achieves its desired set-point, resulting in that the designed ELMO controller and motor model are used in each joint model in 20-sim. The motor parameters and characteristics are adapted for each joint.



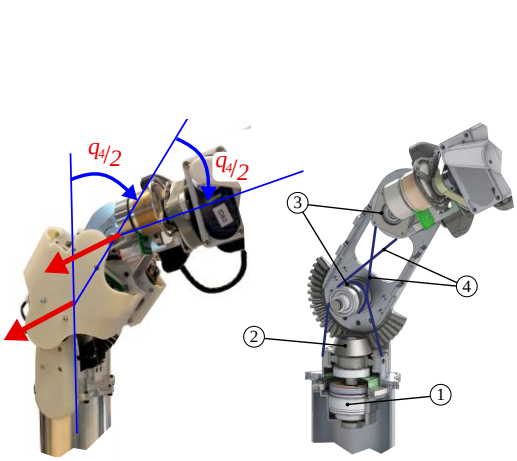
(a) Generated motion profile [Counts]. **(b)** Output of the motor model [RAD].

Figure 3.4: Imitated ELMO control on a motor model

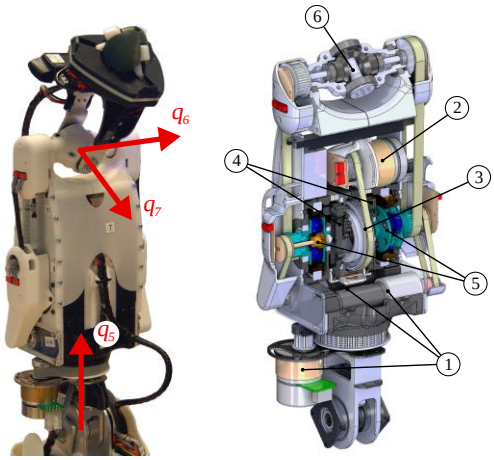
3.3 Elastic Joints: Elbow and Wrist

3.3.1 CAD drawing and 20-Sim models

In figure 3.5 a cad drawing of the elastic joints is presented. The elbow(q4) is actuated with a Robodrive ILM 50-14 in combination with a harmonic drive CSD-25-100-2A-GR-BB. The actuator drives both axes of the elbow. The two axes are connected by means of a steel cable construction and pulley's. The wrist(q5) is actuated in with a Maxon EC45 Flat in combination with a Spur Gearhead GS 45 A gearbox and belt transmission. Besides the motor encoders, two absolute encoders are present. The two absolute encoders measure the output position of the joint q4 and the joint q5.



(a) The 1-DOF elbow joint showing the coupled DOF q4 with two rotation axes and a section view with (1) actuator,(2) bevel gear transmission,(3) pulleys, and (4) tendons. The elbow link is directly actuated through the bevel gear, while the motion of the lower arm is constrained by the tendon mechanism.



(b) The wrist joint and end-effector showing the axes of the 3-DOFs. The section view shows the (1) joint actuators,(2) stiffness adjustment actuator,(3) stiffness adjustment mechanisms,(4) lever mechanism,(5) planetary differential, and (6) bevel differential.

Figure 3.5: CAD drawing of the elbow and wrist (Barrett et al., 2016).

Based on the mechanical design bond-graph models are created and implemented in 20-sim, to be able to perform simulations. The bond-graph model of the wrist(q5) is depicted in figure 3.6. This bond-graph model consist of a motor model with gearbox and belt transmission, ELMO controller and the inertia of the wrist. The inertia's of all models are determined with the CAD drawings, data-sheets and by weighting the components. The compliance of the belt is assumed to be sufficiently small compared to the compliance introduced in the ELMO servo controller, such that it is neglected. For characterisation, the joint is deflected by hand while the ELMO controllers keep the motor output position at zero. The deflection and torque are measured using the absolute encoders and an external force/torque sensor. In 20-sim this is simulated by providing the ELMO Control a set-point of zero. The joint is deflected with a modulated source of flow, this implies a continuous deflection motion in the real set-up. The deflection can directly be obtained by integrating the rotational velocity at the one-junction. The torque is the effort after the modulated source of flow.

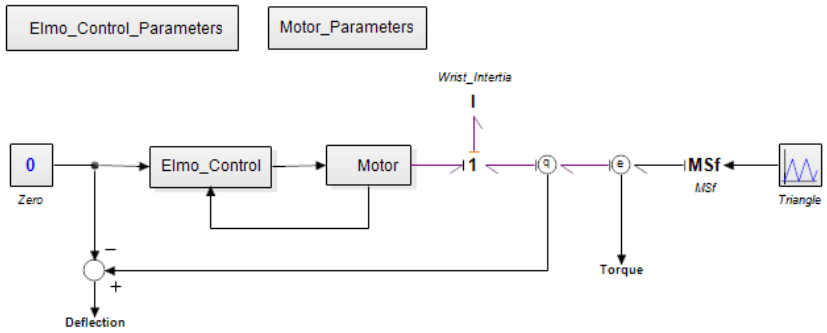


Figure 3.6: 20-sim model of the wrist(q5).

The bond-graph model of the elbow (q4) in 20-sim is depicted in figure 3.7. The model contains a ELMO Control, motor model, upper arm, elbow link and lower arm. The upper arm is connected to joint q3 in the shoulder. This joint is seen as rigid due to the stiffness of the gearbox. It is assumed that this joint does not move during torque control. This is modelled as a source of flow with zero input connected to the 1-junction Upperarm. In an ideal case there is no friction and compliance and the output rotational velocity at the upper arm can be described as in 3.1.

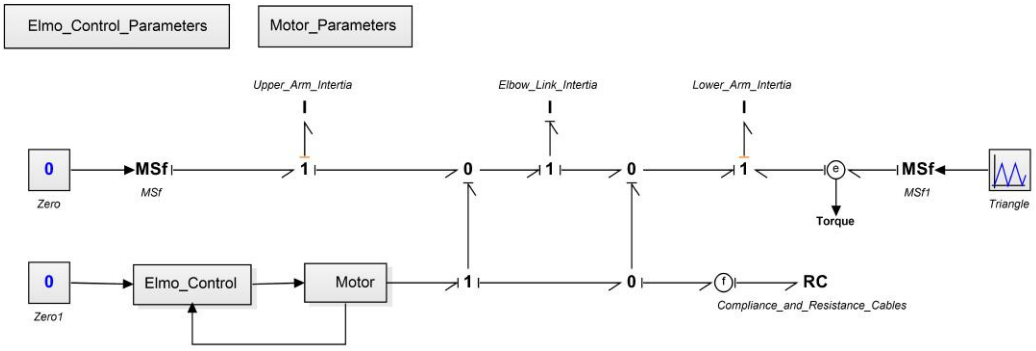


Figure 3.7: 20-sim model of the elbow(q4).

$$\begin{aligned} \dot{\theta_{Out}} &= \dot{\theta_{motor}} + \dot{\theta_{RC}} \\ \dot{\theta_{RC}} &= \dot{\theta_{motor}} \\ \dot{\theta_{Out}} &= 2 * \dot{\theta_{motor}} \end{aligned} \tag{3.1}$$

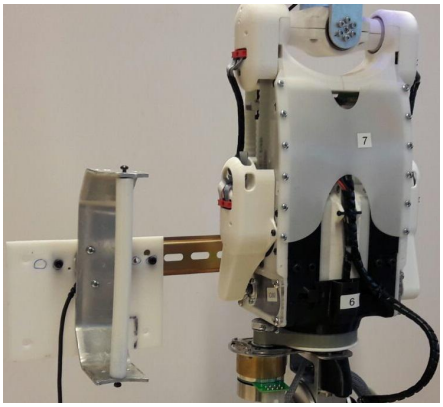
However there is structural friction and compliance in the steel cable construction and pulley's. To identify the creep, play, non-linear friction and hysteresis in this mechanical construction is challenging. It is therefore assumed in this model that this compliance and friction is linear. The friction and compliance is combined in the RC element, the output position is then described as:

$$\dot{\theta_{Out}} = \dot{\theta_{motor}} + \dot{\theta_{RC}} \tag{3.2}$$

For characterisation, the joint is deflected by hand while the ELMO controllers keep the motor output position at zero. The observed deflection and torque are measured with the absolute encoders and an external force/torque sensor. In 20-sim this is simulated by providing the ELMO Control a set-point of zero. The joint is deflected using a modulated source of flow, this implies a continuous deflection motion in the real set-up. The torque is the effort after the modulated source of flow located at the lower arm. The deflection can directly obtained by integrating the rotational velocity towards the RC element.

3.3.2 Characterisation

After constructing the dynamic model of the two elastic joints, the joints were characterised with the proposed method described in the previous section. The test set-up for both characterisations is depicted in figure 3.8.



(a) Test set-up of the wrist (q5).

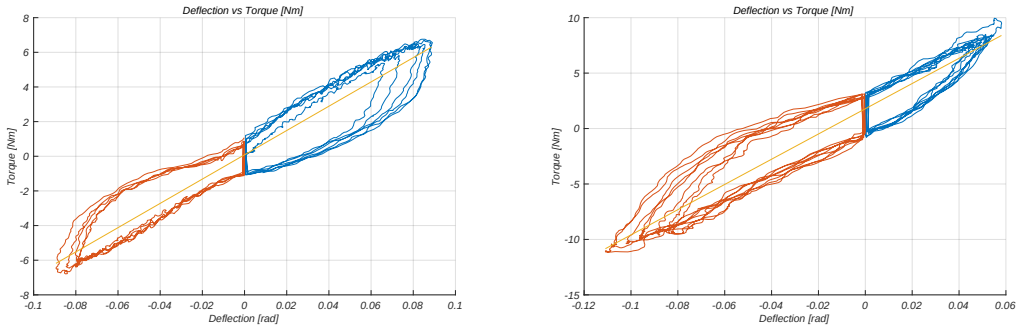
(b) Test set-up of the elbow (q4).

Figure 3.8: Characterisation of the two elastic joints.

Each joint is deflected by applying a force on the handle mounted on the wrist and apply this force with a continuous motion, the achieved joint deflection is determined by the encoders. An Mini40 force/torque sensor from ATI industrial automation (ATI, 2017) is used to measure the applied torque. A matlab script is written to determine the resulted torque deflection characteristics, the results are depicted in figure 3.9.

The torque deflection characteristics indicate that on both joints hysteresis and non linear friction is present. In order to use these torque-deflection characteristics curve fitting is performed to acquire a torque-deflection curve.

Curve fitting is a procedure where a function is used to fit a given set of data in the 'best' possible way without having to match the data exactly. As a result, while the function does not necessarily yield the exact value at any of the data points, overall it fits the set of data well. Curve fitting is normally used when the data has substantial inherent error, such as data gathered from experimental measurements. If the relationship between the independent and de-



(a) Torque-Deflection characteristics wrist (q5). **(b)** Torque-Deflection characteristics elbow (q4).

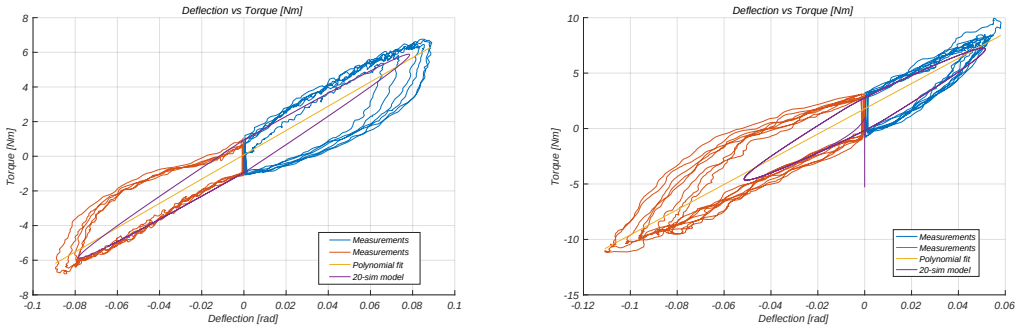
Figure 3.9: Torque-Deflection characteristics of the two elastic joints. The red lines represent the negative deflection, the blue lines the positive deflection and the yellow line the fitted curve.

pendent variables is non-linear, curve-fitting techniques other than linear regression must be used (Chapra, 2006). Curve fitting is achieved with various ways and models, such as regression models, interpolation and smoothing. The regression method contains polynomial, exponential, fourier, gaussian, power, rational, sum of sine and weibullmodels models. The smoothing models are smoothing spline and lowess.

From the acquired data in figure 3.9, conclusion are drawn to determine the best curve fitting technique. In the acquired torque deflection data a hysteresis band is seen and the drop on each side of the is due to the change of direction of the deflection. Resulting in a change of sign in the friction. The friction contains coulomb and viscous friction. In this case there is no dead zone, however in general this could be present. To determine the best fit it is chosen to fit a curve that is in the middle of the hysteresis band. With this method it is possible to add an additional friction model if necessary, to make the fit more accurate. It is chosen to use a polynomial fit. Polynomials are an excellent choice for an approximating function, because of their simplicity, ease of manipulation and evaluation (Hoffman, 2001). The order of polynomial is determined in such a way that the curve is in the middle of the hysteresis band and that higher order polynomials do not sufficiently improve the fit. In order to determine the stiffness the torque deflection curve is differentiated with respect to the deflection, with polynomials this is straight forward and easy to implement in the high-level software in ROS.

To determine the best polynomial fit, the curve fitting toolbox in Matlab is used. A way to determine the goodness of fit is the root-mean-square error. It provides a measure of the differences between values predicted by a model or an estimator and the values actually measured. However this method is sensitive to outliers and is the most common concern with the use of this method (Chai and Draxler, 2014). Matlab provides the root-mean-square error for each fit, however the acquired dataset contains outliers and the root-mean-square error does not represent a good fit. Therefore the order of polynomial is chosen in such a way that it is in the middle of the hysteresis band and represents the possible dead-zone in the best way. For each data set a separate fit for the positive and negative deflection is made as well as a continuous fit for the whole dataset. For these two joints a continuous polynomial fit of the first order is the best fit. This results in a stiffness for the elbow (q4) of 113.78 $[\frac{Nm}{RAD}]$ and for the wrist (q5) of 70.0966 $[\frac{Nm}{RAD}]$. The fitted curve is depicted in figure 3.9

With the acquired torque deflection curve the dynamic models are extended and the model is verified. The same characterisation experiment is repeated with the dynamic models in 20-sim. The results are presented in figure 3.10. In case of the wrist joint q5 there is no direct compliant element, however the compliant behaviour is linked to the proportional gain of the ELMO controller. The 20-sim model of the elbow is compared with the measurements until the maximum positive measured deflection is reached.



(a) Torque-Deflection characteristics wrist (q5). **(b)** Torque-Deflection characteristics elbow (q4).

Figure 3.10: Torque-Deflection characteristics wrist (a) and elbow (b) in 20-sim compared to the measurements. The red lines represent the negative deflection and the blue lines the positive deflection.

The presented 20-sim models with the implemented torque deflection characteristics in figure 3.10 verify that the dynamic models contain the same behaviour as the measured torque deflection characteristics on the actual set-up. However the friction models in the dynamic models of both joints do not model all aspects. These friction models only consist of coulomb friction, however in the real set-up there is coulomb and viscous friction and possible non-linear friction due to the mechanical structure.

3.4 Variable stiffness Joint: Shoulder

The elastic joints are characterised and modelled, the remaining joints are the two VSA joints. The first joint is the shoulder (q2).

3.4.1 CAD drawing and 20-Sim Models

In figure 3.11 a cad drawing of the VSA joint shoulder is presented. The shoulder is actuated with a Robodrive ILM85-13 in combination with a harmonic drive CSD32-100. The actuators are coupled with a differential drive to actuate q2 and q3. The differential drive is constructed by means of a steel cable construction and pulley's, resulting in the following transmissions from actuators a_i to joints q_i (Barrett et al., 2016):

$$\begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{1}{2} & -\frac{1}{2} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \tag{3.3}$$

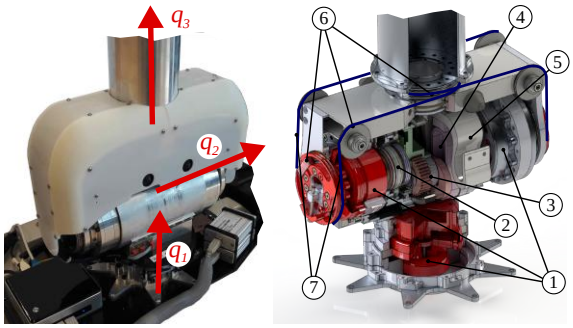


Figure 3.11: The 3-DOF shoulder joint with variable stiffness mechanism showing the DOFs and (1) joint actuators, (2) stiffness adjustment actuators, (3) stiffness adjustment mechanism, (4) lever mechanism with (5) spring, (6) pulleys, and (7) tendons (Barrett et al., 2016).

In addition the shoulder is equipped with a VSA dedicated for q2. The output stiffness can be adjusted by varying q, as explained in chapter 2, with an actuator. The shoulder contains 6 absolute encoders and incremental encoders for each motor. One absolute encoder located at the output gearbox to determine the output position for joint q1. Two absolute encoders, each located on the pulley of the differential drive. One absolute encoder to determine the pivot position of the VSA, located at the VSA. One absolute encoder located inside the shoulder, which measures the deflection of the joint directly. The last encoder is located at the left side of the differential drive and measures the position output of the joint q2.

During the torque control measurements the arm is in contact and therefore the position of the joint does not change. So the internal deflection is only measured by the absolute encoder that measures the deflection directly or using the incremental encoders on the motors. At first the incremental encoders were used, however the CAN bus interface contains communication problems. Which results in loss of data (encoder value's of zero), see appendix A for an elaboration of this problem. Therefore the absolute encoder that measures the deflection directly is chosen to determine the deflection. It is possible that the shoulder starts in a non zero angle. Therefore the initial angle of the shoulder joint(q2) is subtracted from the measured deflection. In free movement the encoders contain the same angle and the resulted deflection is zero. With this method the deflection is correctly determined in each initial position of the shoulder.

From this cad drawing a bond-graph model is constructed and implemented in 20-sim. The bond-graph model of the shoulder (q2 and q3) is depicted in figure 3.12. This bond-graph model consist of two motor models with a gearbox, two ELMO controllers, a pivot mechanism, the differential coupling and the inertia of q3 and q2. The friction and compliance of the VSA is combined in the RC element. The compliance in the tendons and stick-slip friction is not modelled, since this non-linear behaviour is challenging and time consuming to determine. The friction and compliance in the mechanical structure is assumed to be linear and is combined in the RC element.

The VSA and its variable stiffness mechanism is not modelled, instead each pivot position can directly be set. In practice the ELMO will reach its desired position and thus its pivot position. Adding these sub-models will make the model more complex and does not contribute to the behaviour of the joint. It is assumed that the lever in the pivot mechanism does not deflect and is rigid.

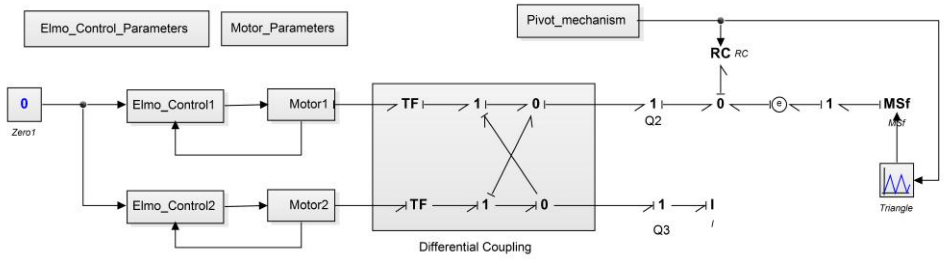


Figure 3.12: 20-sim model of the Shoulder (q2)

For characterisation, the joint is deflected by hand while the ELMO controllers keep the motor output position at zero. The deflection and torque are measured using the absolute encoders and an external force/torque sensor. In 20-sim this is simulated by providing the ELMO control a set-point of zero. The joint is deflected by using a modulated source of flow, this implies a continuous deflection motion in the real set-up. The deflection is directly obtained by integrating the state of the RC element. The torque is the effort after the modulated source of flow.

3.4.2 Characterisation

After constructing the dynamic model of the shoulder, the joint is characterised using the proposed method described in the previous section. The test set-up for the characterisation is depicted in figure 3.13.

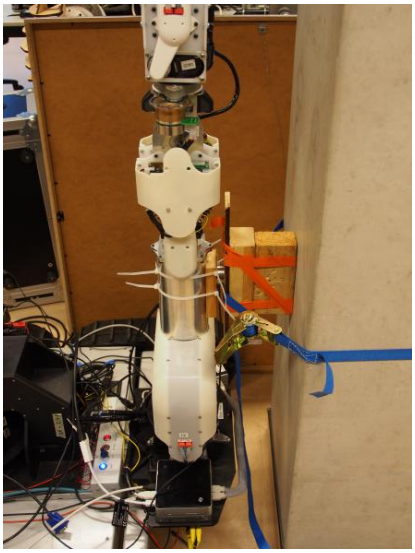
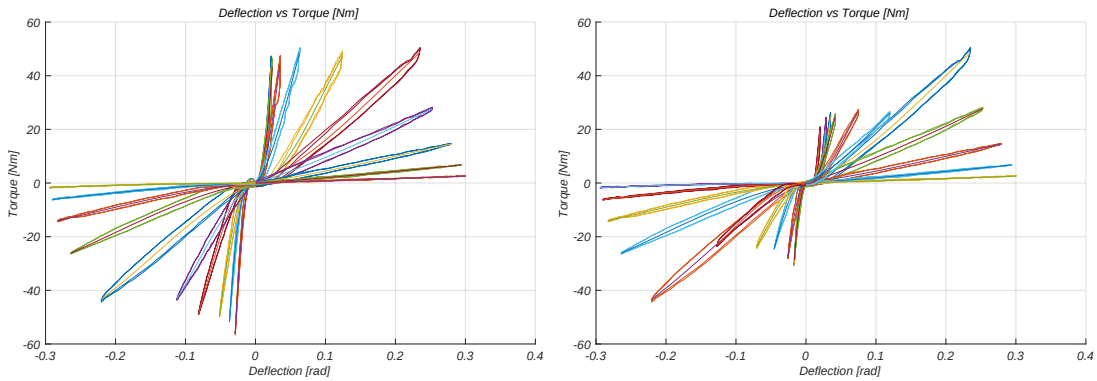


Figure 3.13: Characterisation of the shoulder joint (q2).

The shoulder is fixed on a beam such that it was not possible to move. Each joint is deflected by actuating the shoulder joint (q2) with a sine position trajectory. Resulting in a continuous motion, the achieved joint deflection is determined by the absolute encoder. An Mini40 force/torque sensor from ATI industrial automation is used to measure the force/torque. The force/torque sensor is placed between the robotic arm and the wooden blocks that make contact with the beam. It is chosen to change the stiffness of the VSA linear over the pivot position. In this case from 0 [mm] to 45 [mm] with intermediate steps of 5 [mm], resulting in 10 stiffness settings. For the pivot position 0 till 30 [mm] a small and larger deflection trajectory is used. The larger deflection trajectories are used to determine the performance of the shoulder. For safety reasons the smaller deflection trajectories are used to limit the output torque. A Matlab script is written to determine the resulted torque deflection characteristics, the results are depicted in figure 3.14.



(a) Torque-Deflection characteristics with large deflections.

(b) Torque-Deflection characteristics with small deflections.

Figure 3.14: Torque-Deflection characteristics of the shoulder joint (q2).

The torque deflection characteristics clarify that on both joints hysteresis, dead-zone and non-linear friction is present. At lower stiffness settings there are vibrations in the mechanical structure, resulting in vibrations in the measurement data. This data is smoothed with a moving average filter of 8 samples. In order to use these torque-deflection characteristics curve fitting is performed to acquire a torque-deflection curve. For each data set a separate fit for the positive and negative deflection is made as well as a continuous fit for the whole dataset. For the shoulder a separate fit for the positive and negative deflection of a third degree polynomial is the best fit. The fitted curves are presented in figure 3.9. The coefficients of the polynomial curve are forced to provide zero torque at zero deflection. The dead-zone behaviour is represented and the curve is in the middle of the hysteresis band.

In order to determine the torque-deflection characteristics for all possible pivot positions, linear interpolation between the measured pivot positions is performed. For each pivot position the torque of the two nearest measured pivot positions are calculated and an weighted average of these torques is taken. In figure 3.15 the fitted curves for the pivot positions 0 [mm] till 45 [mm] with intermediate steps of 2.5 [mm] is depicted.

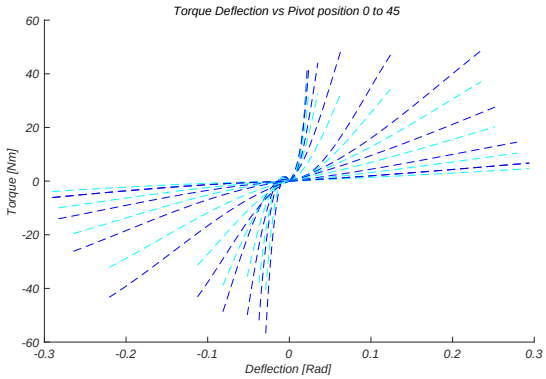


Figure 3.15: Fitted third degree polynomial for the pivot positions from [0 mm] to 45 [mm] with intermediate steps of 2.5 [mm]. The blue curves represent the fitted curve for the measured pivot positions. The magenta curves the linear interpolation between the measured pivot positions.

With the acquired torque deflection curve the dynamic model is extended and the model is verified. The same characterisation experiment is repeated with the dynamic models in 20-sim. The results are presented in figure 3.16. The stiffness is obtained by differentiation the torque deflection curve with respect to the deflection. For each pivot position the stiffness and the resistance is changed in the rc element with the corresponding torque-deflection curve. The input of the source of flow is changed for each stiffness setting to obtain the required deflection.

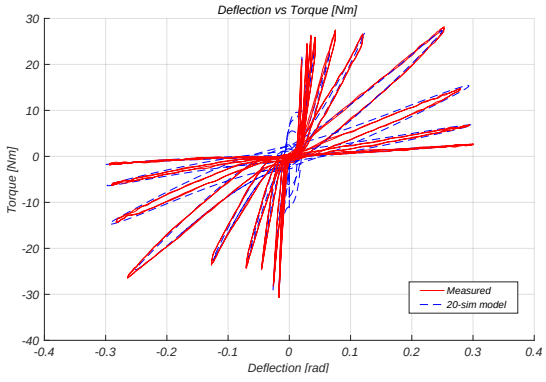


Figure 3.16: Torque-Deflection characteristics of the shoulder (q2) in 20-sim compared to the measurements.

The presented 20-sim models with the implemented torque deflection characteristics in figure 3.16 verify that the dynamic models contain the same behaviour as the measured torque deflection characteristics on the actual set-up. Similar as the elastic joints, the friction models in the dynamic model do not model all aspects. This could be improved by adding a non-linear friction model of the tendons and pully's along with identification of the non-linear friction due to the mechanical structure.

3.5 Variable stiffness Joint: Wrist

The last joint that is modelled and characterised is the 2-DOF VSA joint wrist(q6 and q7).

3.5.1 CAD drawing and 20-Sim Models

Overview of the model

In figure 3.17 a CAD drawing and schematic overview of the internal components of the VSA joint wrist is presented. The wrist (q6 and q7) is actuated with two Maxon RE25 motors in combination with GP-32-C gearboxes. A coupled stiffness adjustment mechanism is implemented in the wrist, requiring only one Maxon actuator to adjust the output stiffness in both joints $q6$ and $q7$. A planetary gear system is used as 3-port differential in order to incorporate the variable stiffness mechanisms into the drive train in this way. The levers, i.e. the output of the variable stiffness modules, are connected to the ring gears and the driving actuators to the sun gear, such that the planet carrier functions as output. The two drivetrains are then in turn differentially coupled to each other through a bevel gear mechanism, to actuate the DOFs q6 and q7 (Barrett et al., 2016).

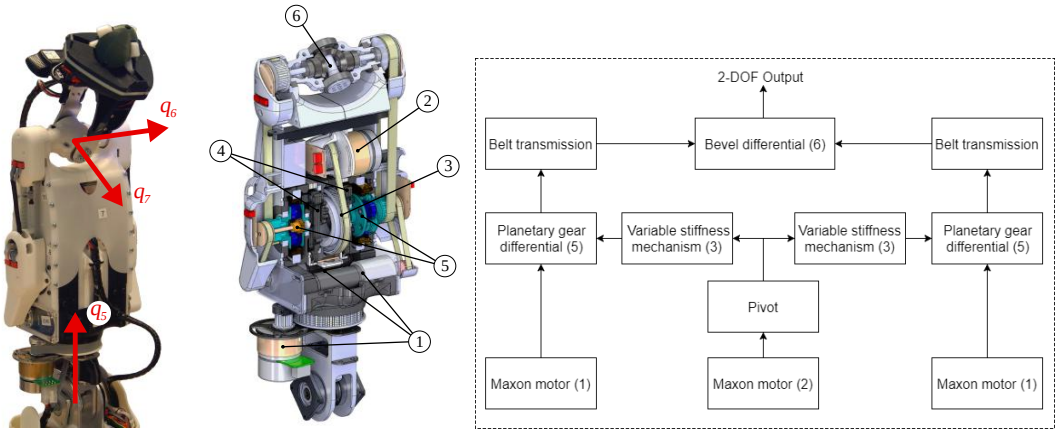


Figure 3.17: The wrist joint and end-effector showing the axes of the 3-DOFs. The section view shows the (1) joint actuators, (2) stiffness adjustment actuator, (3) stiffness adjustment mechanisms, (4) lever mechanism, (5) planetary differential, and (6) bevel differential (Barrett et al., 2016).

The wrist contains 6 absolute encoders and incremental encoders on each motor. Two absolute encoder inside the wrist to measure the output position of q5 and the deflection of the wrist joint (q6 and q7) . Two absolute encoders on each planetary gear that measure the input of the planet carier (gearbox output). One absolute encoder to determine the pivot position of the VSA. Two absolute encoders on each side of the bevel differential input.

From this cad drawing a bond-graph model is constructed and implemented in 20-sim in order to perform simulations. An overview of the bond-graph model of the wrist (q6 and q7) is depicted in figure 3.18. This bond-graph model consist of two motor models, one for each actuated DOF. Inside each motor model a bond-graph of the motor model including gearbox and a ELMO controller is present. The VSA and its variable stiffness mechanism is not modelled, instead each pivot position can directly be set. In practice the ELMO will reach its desired

position and thus its pivot position. Adding these sub-models will make the model more complex and does not contribute to the behaviour of the joint. It is assumed that the lever in the pivot mechanism does not deflect and is rigid. The compliance of the VSA is modelled as a C-element. Besides the two motor models and the pivot mechanism, the 20-sim model of the wrist contains two planetary gears, two belt transmissions and a bevel differential. Each sub-model is explained in the sections below, starting from the planetary gear towards the bevel differential.

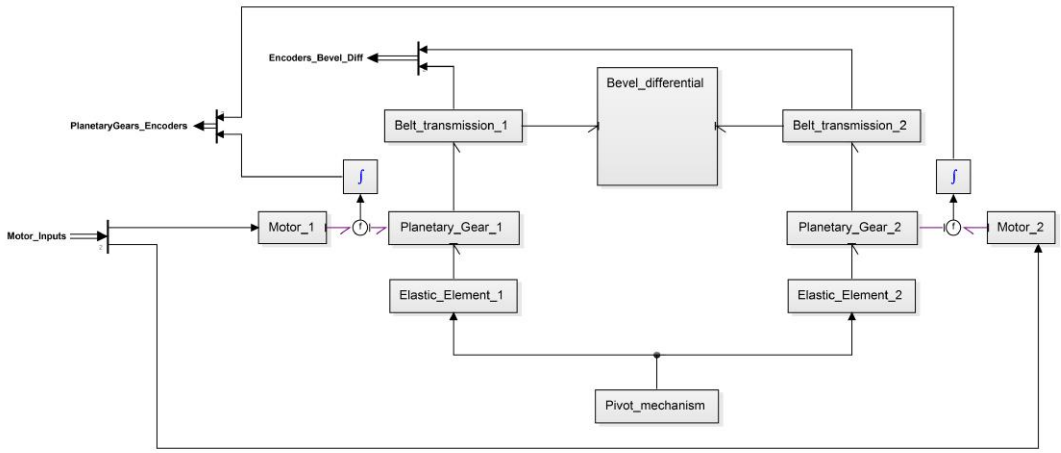


Figure 3.18: 20-sim model of the wrist (q6 and q7).

Planetary Gear differential

In order to combine the motion of the primary actuator, the elastic elements and the output, a 3-ports transmission is required. This is accomplished by a planetary gear differential, an overview of the planetary gear differential is presented in figure 3.19.

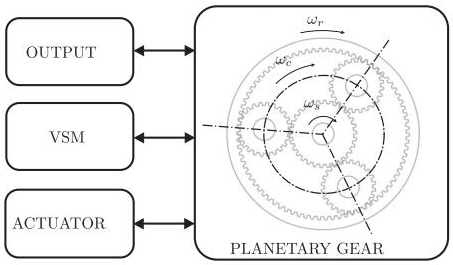


Figure 3.19: Schematic overview of the planetary gear set (Tan et al., 2017).

There are two planetary gear differentials, one for each DOF. The two primary actuators are connected to the planetary sun gears, ω_s. The VSM connects to the ring gears, ω_r. The output is then connected to the planet carriers, ω_c. (Tan et al., 2017) In (Tan et al., 2017) and (Zult, 2016) equations for the planetary gears are derived. The rotational velocity of the output of the planetary gear is described as:

$$\begin{aligned} (N_r + N_s) * \omega_c &= N_r * \omega_r + N_s * \omega_s \\ \omega_c &= \frac{1}{1 + \frac{N_s}{N_r}} * \omega_r + \frac{1}{1 + \frac{N_r}{N_s}} * \omega_s \end{aligned} \tag{3.4}$$

Where N_s and N_r are the number of teeth in the ring and the sun gear respectively. Note that the number of teeth of a planet carrier gear is not a separate required parameter as it is geometrically determined by N_s and N_r. These relations are transformed in a bond-graph model which is depicted in figure 3.20. A zero-junction provides the sum of the two flows, which come from the ring velocity and the sun velocity. Both flows go through a transformer of which the parameters are described as the derived equations in 3.4.

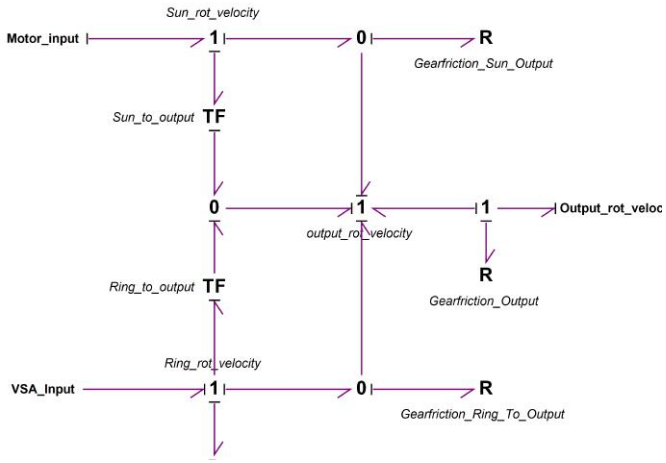


Figure 3.20: Overview of the planetary gear set in 20-sim.

The gear-friction was determined in (Zult, 2016) and contains viscous and coulomb friction. The friction is described in 3.5 with an arctan-type fuction.

$$F_{gear}(\delta\omega) = k_a * \arctan(k_b * \delta\omega) \tag{3.5}$$

Zult (2016) has shown that the continuous arctan type friction model is competent enough. It is possible to extend this friction model with a more complex model. The arctan-type friction model contains two parameter which are determined experimentally.

Belt transmission

The output of each planetary gear is connected to a belt transmission with a ratio of 30:40. An overview of the belt transmission is presented in figure 3.21. The ratio of the belt transmission is implemented in the transformer. The same friction model as in the planetary gear is implemented to determine the mechanical friction in the belt. The parameters are determined experimentally. The compliance of the belt is assumed to be sufficiently small compared to the compliance introduced from the VSA, such that it is neglected.

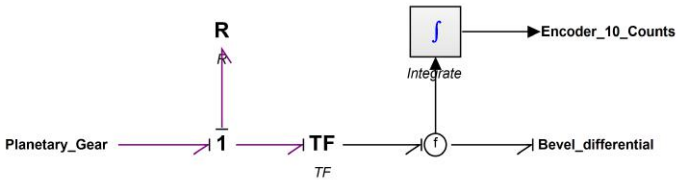


Figure 3.21: 20-sim model of the belt transmission in the wrist

Bevel differential

A cross section of the the bevel differential is presented in figure 3.22.

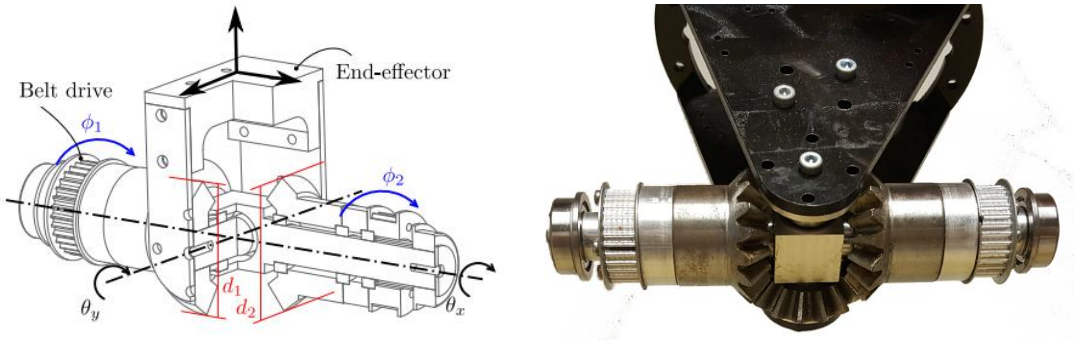


Figure 3.22: Cross section of the bevel differential.

The outputs of the belt transmissions are connected to a bevel differential which transforms the rotations ϕ_1 and ϕ_1 to rotations θ_x and θ_y around the x- and y-axes in Cartesian workspace (Tan et al., 2017).

$$\begin{pmatrix} \dot{\theta_x} \\ \dot{\theta_y} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ \frac{d_1}{d_2} & -\frac{d_1}{d_2} \end{pmatrix} \begin{pmatrix} \dot{\phi_1} \\ \dot{\phi_2} \end{pmatrix} \tag{3.6}$$

In this equation d_1 and d_2 are the gear diameters. Both gear sizes are identical, $d_1 = d_2$, which results that the rotations around the x- and y-axes in Cartesian workspace are describe as:

$$\begin{aligned} \dot{\theta_x} &= \frac{1}{2}(\dot{\phi_1} + \dot{\phi_2}) \\ \dot{\theta_y} &= \frac{1}{2}(\dot{\phi_1} - \dot{\phi_2}) \end{aligned} \tag{3.7}$$

With the derived relationships a bond-graph model is constructed and the model is depicted in figure 3.23. In this model one zero-junction is used to get the sum of velocities, while the other provides the difference of velocities between $\dot{\phi_1}$ and $\dot{\phi_2}$. A transformer is used to add the factor of $\frac{1}{2}$. The same friction model as in the planetary gear is implemented to determine the mechanical friction in the gears. The parameters are determined experimentally.

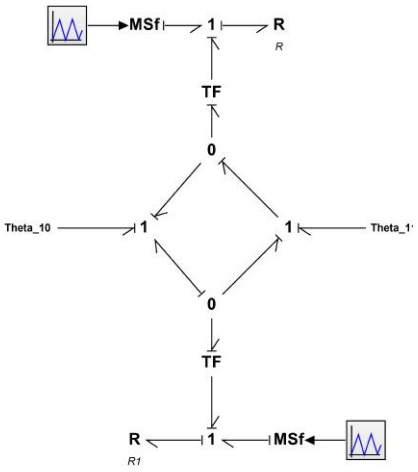


Figure 3.23: 20-sim model of the bevel differential.

Deflection

The deflection of the wrist joint (q6 and q7) is determined with four absolute encoders. Two encoders on each planetary gear that measure the input of the planet carrier (gearbox output), located at 5 in figure 3.17. The other two absolute encoders are the encoders on each side of the bevel differential input, located at 6 in figure 3.17. It is possible, using the direct kinematics derived in the previous sections, to link the encoder at the input of the planet carrier to the encoder located at the bevel differential. It is assumed that the input of the ring gear in the planetary gear is fixed and does not change during actuation of the wrist joint (q6 and q7). The relationship is described in 3.8. The two encoder value's are subtracted from each other. With this method the two encoders have the same angle in free movement, which results that the deflection is zero. In case the arm is fixed or in contact with its environment (torque control) the encoder at the bevel differential is not able to rotate, however the deflection can still be measured with the encoders located at the input of the planet carrier. The initial position that the wrist joint contains is then taken into account. To determine the output angle in Cartesian workspace the equations in the previous section are used. In this equation θ_a is the encoder of the bevel differential, $\frac{D_p}{D_b}$ is the ratio of the belt transmission between the planetary gear and bevel differential, θ_{pc} is the encoder of the planetary gear.

$$\widetilde{\theta_a} = \frac{N_s}{(N_s + N_r)} * \frac{D_p}{D_b} * \theta_{pc}$$
$$Deflection = \widetilde{\theta_a} - \theta_a \tag{3.8}$$

Now that the kinematic structure of the wrist joint (q6 and q7) is described and the deflection of the joint can be determined, the characterisation of the joint is the next step. For characterisation, the joint is deflected by hand while the ELMO controllers keep the motor output position at zero. The deflection and torque is measured using the absolute encoders and an external force/torque sensor. In 20-sim this is simulated by providing the ELMO control a set-point of zero. The joint is deflected by using two modulated source of flows in the bevel differential, this implies a continuous deflection motion in the real set-up. The deflection is determined by integration the angular velocities in the 20-sim model at the exact locations as in the real set-up. The relationship of the direct kinematics and the bevel differential are used to determine the deflection in Cartesian workspace. The torque for each joint is the effort after the modulated source of flows.

3.5.2 Characterisation

After constructing the dynamic model of the wrist, the joint is characterised with the proposed method described in the previous section. The test set-up for the characterisation is presented in figure 3.24.

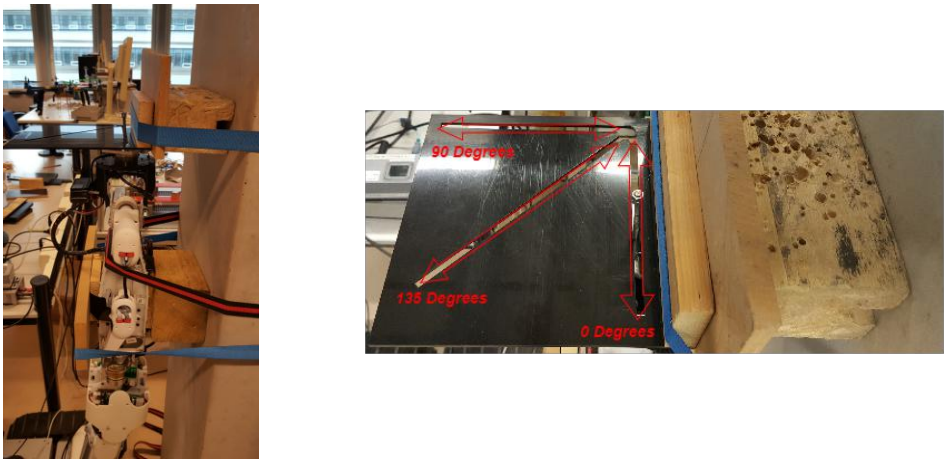
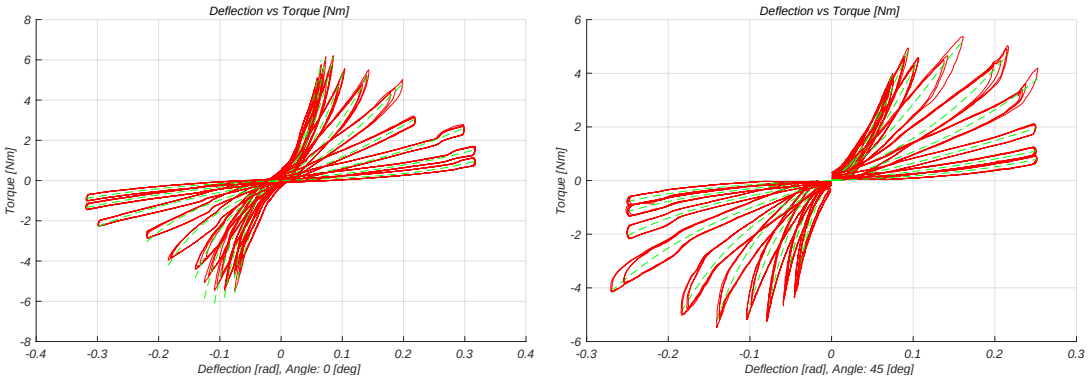


Figure 3.24: Characterisation of the wrist joint (q6 and q7).

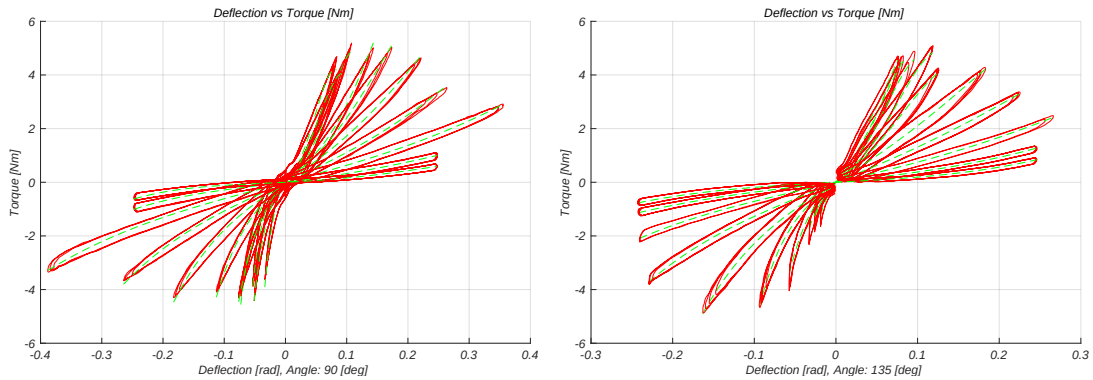
The wrist and arm are fixed to a beam such that it was not possible to move and only internal movement of the wrist joints q6 and q7 are allowed as depicted in figure 3.24. Using this configuration gravity does not influence the measurement. Initially a set-up is used in which the elbow is rotated 90 degrees and the wrist is clamped on the table. However these measurements were influenced by gravity. Since the wrist joint is 2 DOF it is deflection on 4 predefined angles in Cartesian workspace as seen in figure 3.24. These angles are 0 (only q7),45 (between q6 and q7),90 (only q6) and 135 (between q6 and q7). An Mini40 force/torque sensor from ATI industrial automation is used to measure the applied torque. A mounting plate is constructed such that the force/torque sensor is placed on top of the wrist joint replacing the gripper. It is chosen to change the stiffness of the VSA linear over the pivot position. In this case from 0 mm to 50 mm in intermediate steps of 5 mm, resulting in 11 stiffness settings. For each stiffness setting the four angles of deflection are measured separately. A matlab script is written to determine the resulted torque deflection characteristics of each angle of deflection, the results are depicted in figure 3.25 and 3.26.



(a) Torque-Deflection characteristics with 0 degrees deflection.

(b) Torque-Deflection characteristics with 45 degrees deflection.

Figure 3.25: Torque-Deflection characteristics of the wrist joint (q6 and q7) for 0 and 45 degrees deflection.



(a) Torque-Deflection characteristics with 90 degrees deflection.

(b) Torque-Deflection characteristics with 135 degrees deflection.

Figure 3.26: Torque-Deflection characteristics of the wrist joint (q6 and q7) for 90 and 135 degrees deflection.

The torque deflection characteristics clarify that on both joints hysteresis, dead-zone and non linear friction is present. At lower stiffness settings there are a small vibrations in the measurement data, this data is smoothed with a moving average filter of 5 samples. In order to use these torque-deflection characteristics curve fitting is performed to acquire a torque-deflection curve. For each data set a separate fit for the positive and negative deflection is made as well as a continuous fit for the whole dataset. For the wrist (q6 and q7) a continuous fourth order polynomial fit is the best result. The coefficients of the polynomial are forced to provide zero torque at zero deflection. The dead-zone behaviour is represented and the curve is in the middle of the hysteresis band. Between the measured deflection angles and different pivot positions linear interpolation is used to determine the torque-deflection characteristics. The fitted curves are depicted in figure 3.25 and 3.26. Another approach that was examined is to fit a surface based on the four fitted curves for angle and each stiffness setting. This would result in 11 surfaces. Between the surfaces and thus the measured pivot positions, linear interpolation is used. This method has the advantages that it is simple to implement in the high level software. However this approach resulted in a poor fit, the approach is elaborated in appendix B.

With the acquired torque deflection curve the dynamic model is extended and the model is verified. The same characterisation experiment is repeated in the dynamic model for an deflection angle of 0 (only q7) degrees with an initial angle of the wrist of 0 radians. The results are presented in figure 3.27. The stiffness is obtained by differentiating the torque deflection curve with respect to the deflection. For each pivot position the stiffness is changed in the c element inside the Elastic Element block with the corresponding torque-deflection curve. During characterisation a single torque deflection curve is determined for both elastic elements in the wrist (q6 and q7). Therefore in the 20-sim model each C-element is multiplied by a $\frac{1}{2}$ with the torque deflection curve obtained by the characterisation measurements.

The presented 20-sim models with the implemented torque deflection characteristics in figure 3.27 verify that the dynamic model contains the same behaviour as the measured torque deflection characteristics on the actual set-up. For low stiffness's the 20-sim model has a similar behaviour. However for higher stiffness's the performance can be improved. The fitted torque deflection curves are less accurate because of the a-symmetric shape. Secondly the friction models in the bond-graph models of the both joints do not fully satisfy the behaviour of the set-up. This could be improved by adding a non-linear friction due to the mechanical structure and the stick-slip friction of the planetary gears and bevel differential.

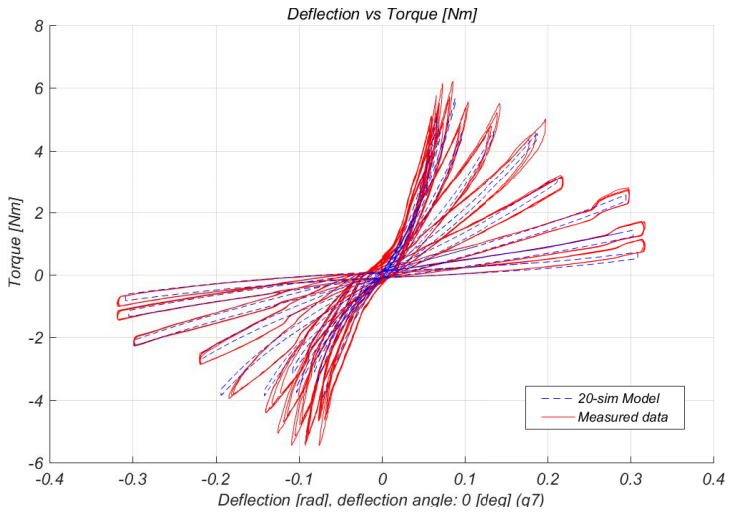


Figure 3.27: Torque-Deflection characteristics of the wrist (q7) in 20-sim compared to the measurements.

3.6 Conclusions

The presented models represent the behaviour of the joints, however improvements can be made by adding the non-linear friction models in the bond-graphs. In the next chapter the acquired torque-deflection characteristics are implemented in the high-level software and a control loop is written to perform torque control. If the performance of torque control is inadequate, the bond-graph models can be extended with non-linear friction models and simulations can be performed to improve the estimated torque.

4 Software implementation

The torque deflection characteristics obtained in chapter 3 are implemented in the observer. The observer provides all the necessary information about the current state of each joint. A controller is written to control the deflection of the two VSA joints (q2,q6 and q7). First a detailed description of the software architecture is provided.

4.1 Software architecture

On the NUC high level control software is implemented in ROS. The control loop runs with a frequency of 40 Hz. In figure 4.1 a detailed overview of the software is presented. The observer provides the high level planner with all the necessary information about the current state of each joint, for example the position. In the high level software various state machines are implemented together with Moveit!. The joint controllers in the mid-level control will execute execute the given path and converts each desired joint position to motor positions. Besides the execution of the trajectory provided by Moveit!, the controller can also execute the set-points from dynamic reconfigure. These set-points are direct joint set-points supplied by a graphical user interface. This allows manual manipulation of the arm. In this graphical user interface it possible to set the joint controllers to execute the Moveit! trajectory or manual manipulation of the arm. The communication messages between these main blocks are:

- The observer provides a joint state message, containing the joint name and position to the high level software and joint controller.
- The high level controller sends the constructed trajectory to the controller.
- The controller will either convert the trajectory generated by Moveit! or the set-points from dynamic reconfigure into motor set-points.

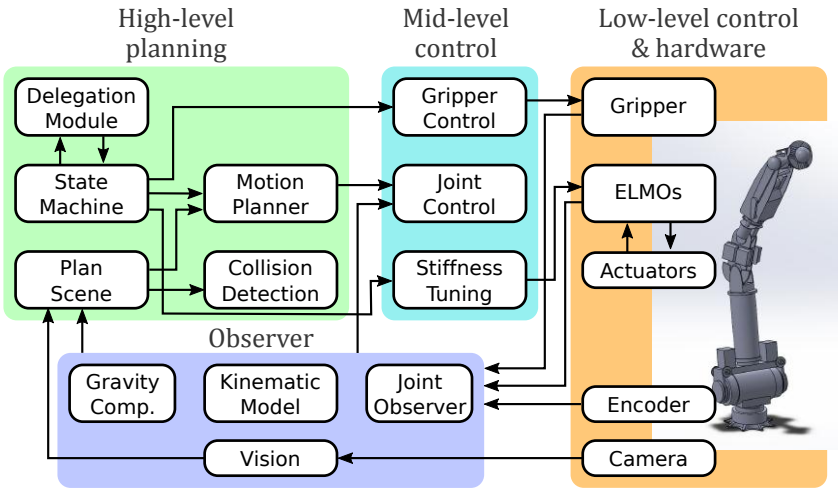


Figure 4.1: Detailed overview of the software architecture

For torque control the observer and the joint state message are extended with the deflection, observed torque and stiffness for each joint. The controller is extended with a third control option, namely torque control. Resulting in that the controller executes either the trajectory from Moveit!, dynamic reconfigure or torque control. The graphical user interface for dynamic reconfigure is extended with torque control to be able to set a desired torque trajectory and to start torque control.

4.2 Observer

The goal of the state observer is to publish the state of each joint. A overview of the observer is presented in figure 4.2. In the constructor the parameters of the arm are loaded form a yaml file. In this configuration file the SPI bus and CAN bus are specified together with the connected absolute encoders, motor drivers (including incremental encoder on the motor) and joint feedback gains and joint limits. This results in that all these topics, such as a absolute encoder position in radians, are generated and available. A list is constructed to access all absolute encoders and motor drives. Once this configuration file is loaded a individual observer for each joint is constructed. Each joint controller is assigned with a unique name and is provided with the constructed encoder and motor drives list. This enables that each joint observer can access all necessary information, such as encoder value's.

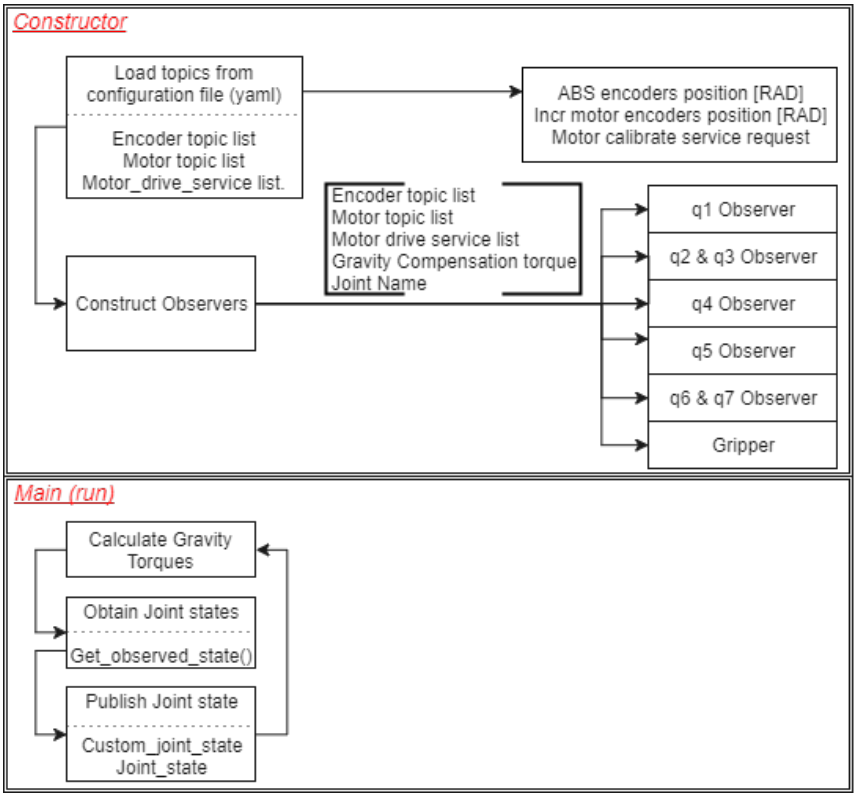


Figure 4.2: Overview of the observer

Once the state observer is constructed, the node loops in the main function. It calculates the joint torques due to gravity, such that these can be used for used for future work. The gravity torques are calculated with a Unified Robot Description Format(URDF) file. This calculation is elaborated in appendix C. Once the gravity torques are calculated, the state of each joint observer with the function get_observed_state is obtained. A detailed description and flow chart of this function is depicted in figure 4.3, which represents the function for the joint wrist (q6 and q7). The shoulder joint (q2) contains a similar approach and contains the red and and blue blocks. There is no conversion made to polar coordinates since it is a 1 DOF joint. Both elastic joints q4 and q5 only consist of the red blocks, since it doesn't contain variable stiffness mechanism and the joints are 1 DOF.

First the stiffness setting and a scale factor is determined. The stiffness setting is determined with the pivot position. The pivot position is calculated based on the encoder value of the variable stiffness mechanism. These stiffness setting correspond with the measured pivot positions. In general the measured pivot position and thus stiffness setting, is in between two of the measured pivot positions linear interpolation is required. The scale factor q determines the

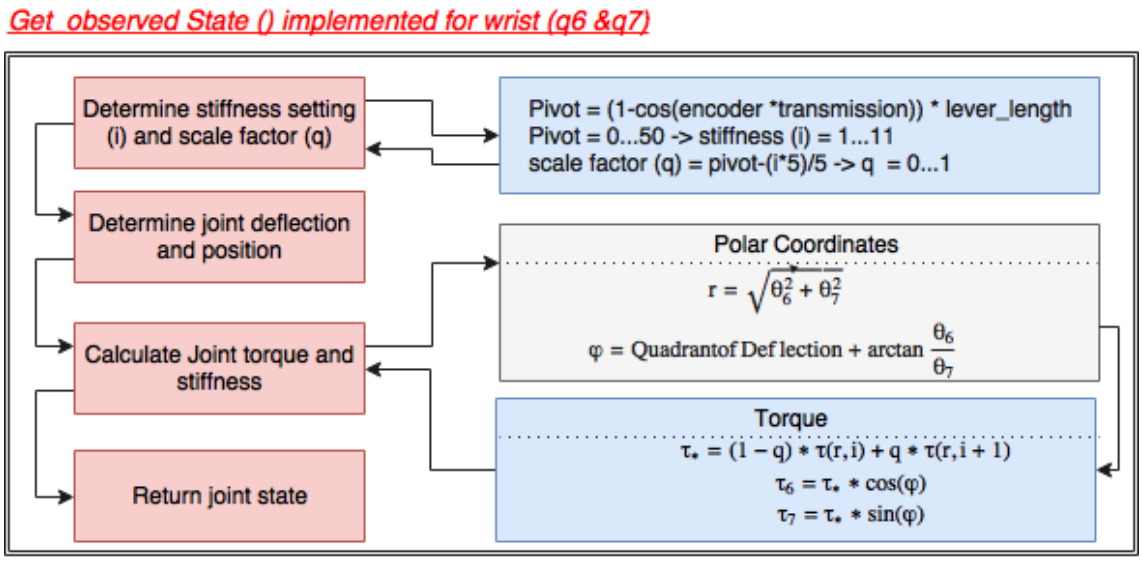


Figure 4.3: Detailed overview of the function get_observed_state.

value between the two measured pivot positions, this is a value between 0 and 1. After determination of the stiffness setting(i) and the scale factor(q), the joint deflection and position are calculated based on the kinematics.

The final step is to calculate the torque and stiffness providing the stiffness setting and the scale factor. In the wrist a switch to polar coordinates is made to r and θ. When determining θ the quadrant in which the deflection is presented is taken into account. The torque in polar coordinates is calculated with two fitted torque deflection curves providing r and the stiffness setting i. Linear interpolation between the two fitted deflection curves is achieved with the scale factor q. The calculated torque is converted from polar coordinates to Cartesian coordinates to obtain the joint torque for q6 and q7 respectively. The same method is applied for the stiffness. Two stiffnesses are calculated and linear interpolation is performed with the scale factor q to determine the stiffness of the joints.

Initially the stiffness was calculated based on the derivative of the torque deflection curve with respect to the deflection. However this resulted in unstable performance in the controller. Therefore the stiffness is now calculated based on a first order fit of the torque deflection curve for each measured stiffness setting. Which results that the derivative of the torque deflection curve with respect to the deflection is a positive fixed number.

The current state of the joint is determined and the function then returns the current position, deflection, stiffness and torque. Once the state of each joint is know, it is published on the ros network with the messages custom_joint_state and joint_state. These messages are used by the controller and contain the current position, deflection, torque and stiffness of each joint. After publishing the joint states the whole cycle is repeated with the stated loop rate of 40 [Hz].

4.3 Controller

The controller is constructed in a similar way as the observer. A overview of the controller is depicted in figure 4.4. In the constructor, joint controllers are created. The joint controllers are provided with a unique name, control mode, joint set-point and drive control mode. These joint controllers convert the joint set-points to motor set-points based on the mechanical structure.

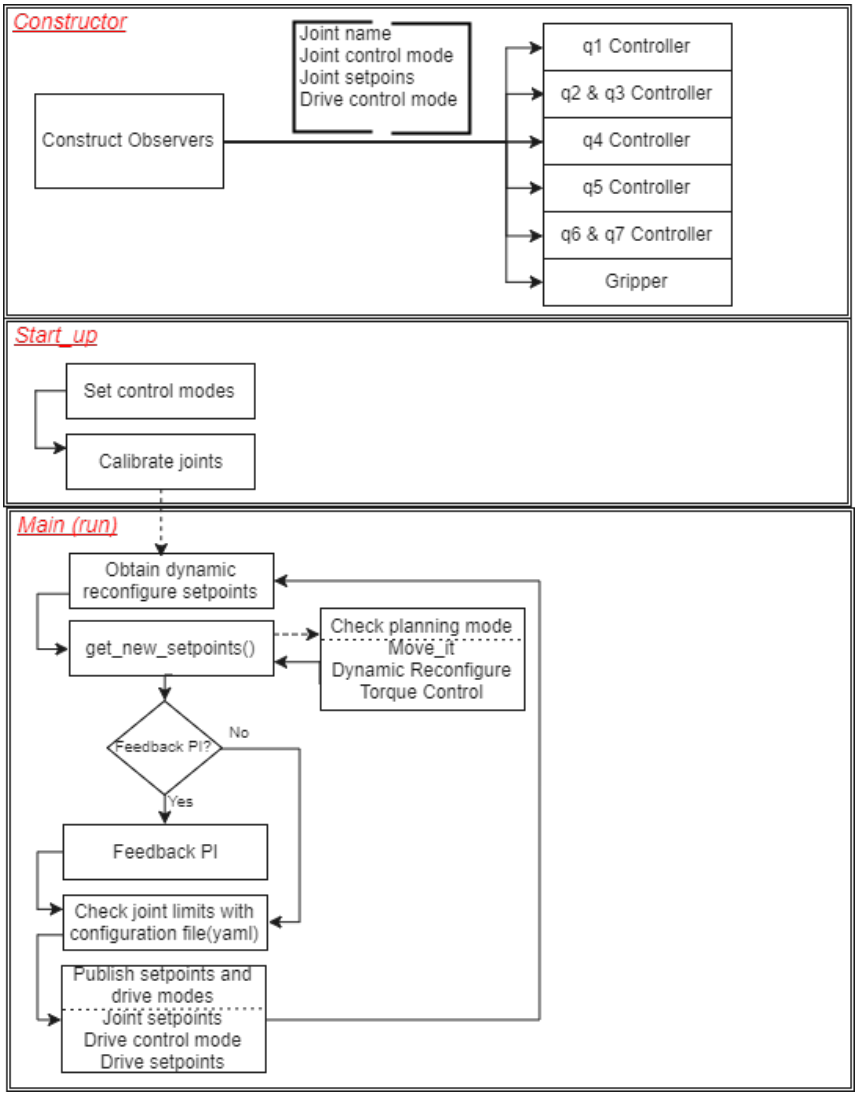


Figure 4.4: Overview of the controller.

Once the joint controllers are created the controller is initialized by setting the control mode for each joint controller. These modes are position, velocity or current control. At this moment only position control is possible since the ELMO servo controllers only have position control implemented. However the structure to extend to velocity or current control is present. After setting the control mode the joints are calibrated and the arm moves in home position. The required joint set-points for calibration are set in the configuration file (yaml).

The controller then loops in the main function. It obtains the set-points from dynamic reconfigure which are set by the graphical user interface. The graphical user interface is extended for the torque control and a desired desired torque trajectory can be set for the joints q2, q6, q7. The graphical user interface is presented in figure 4.5.

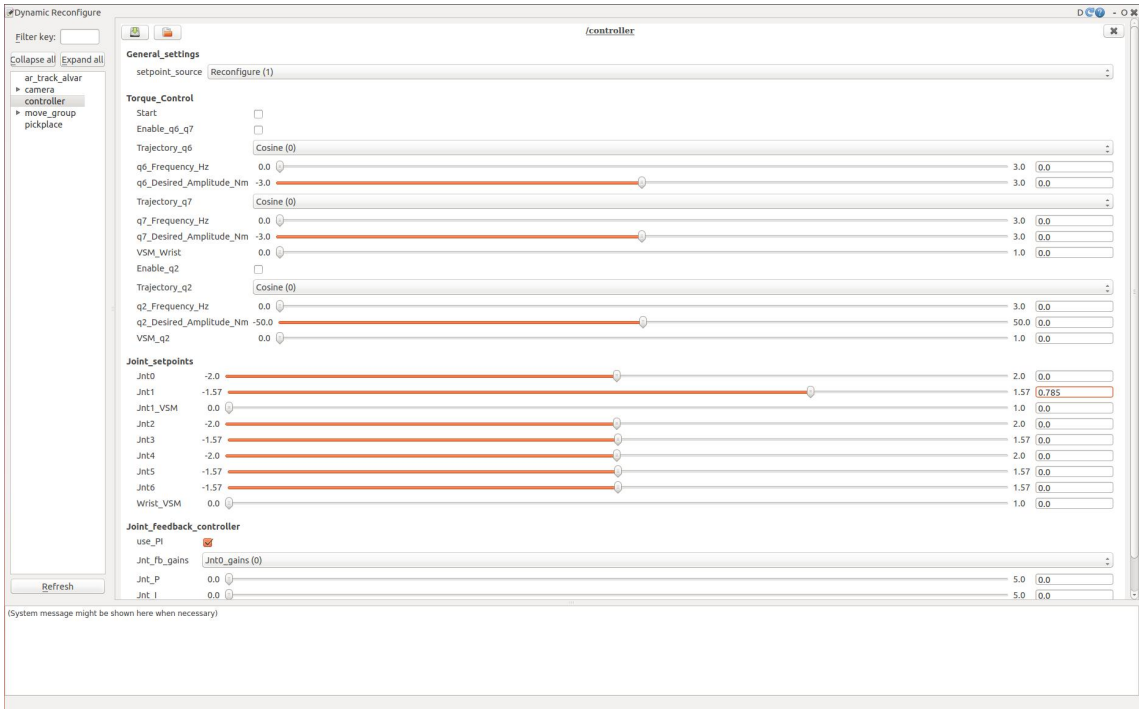


Figure 4.5: Graphical user interface of dynamic reconfigure.

After obtaining the set-points from dynamic reconfigure the function get_new_setpoints is called. This contains three options based on the given control mode. Either joint setpoints from the Moveit! package, dynamic reconfigure (manual manipulation) or torque control are obtained. The torque control mode is written to be able to perform torque control for the joints q2, q6 and q7. A block diagram of the control scheme for torque control is presented in figure 4.6.

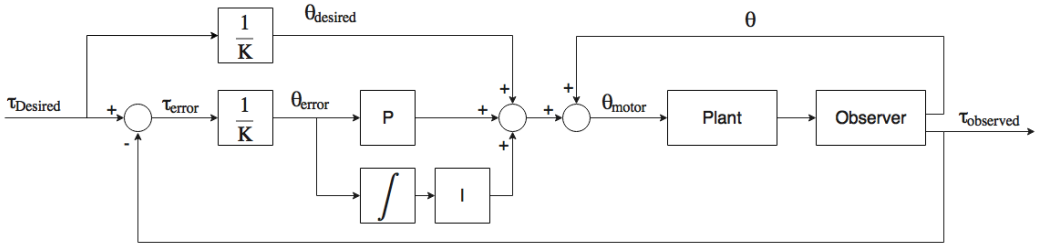


Figure 4.6: Block diagram of the control scheme for torque control.

In order to perform torque control, the controller is provided with the current joint state from the observer. A torque error is calculated based on the required torque set-point set in the dynamic reconfigure graphical user interface and the observed joint torque. The torque error is divided by the stiffness to obtain a deflection error. The deflection error is fed to a PI loop. The gains of the PI loop are tuned experimentally. The output of the PI loop is summed with a feedforward deflection and the current joint position term. This results in a new joint set-point.

After obtaining the joint set-points a check is performed if a feedback PI loop is activated. This PI control loop is provided with the current state and the PI loop ensures that the given joint set-point is obtained. In case of the control mode Moveit! or dynamic reconfigure (manual manipulation) the feedback PI loop is by default activated. In case the control mode is torque control the feedback PI loop is deactivated since it interferes with the torque controller. The joint set-points are checked such that they do not exceed the joint limits. These limits are de-

fined in the configuration file (yaml). The joint set-points are published on the ros network. After publishing the joint set-points it returns to obtaining the dynamic reconfigure set-points and the whole cycle is repeated with a loop rate of 40 [Hz].

Torque control is achieved by determining the current joint state in the state observer. The current joint state is provided to the controller. The controller obtains the desired torque trajectory from the dynamic reconfigure graphical user interface. In the controller the joint set-points are calculated with the control mode, torque control. The joint controllers that were created in the constructor obtain these new joint set-points and convert the desired joint positions to motor positions. These set-points are sent to ELMO's using the CAN bus. These will perform local position control to achieve the given set-point.

5 Experiments and validation

The implemented torque control is validated for the shoulder joint (q2) and the wrist joints (q6 and q7). For various stiffness settings a reference torque is set in the graphical user interface. The chosen torque trajectories are a sine, cosine and a step function. The amplitudes of these functions are based on the acquired torque during characterisation and are determined by the stiffness setting. The stiffer the joint is set, the higher the amplitude of the torque trajectory. The frequency for the sine and cosine are 0.1 and 0.2 Hz. Higher frequencies resulted in poor performance due to the 40 Hz control loop and the latency between the high level software and the CAN and SPI Bus. The controller is then no longer able to track the desired torque trajectory. A comparison is made between the reference torque, a Mini40 force/torque sensor from ATI industrial automation to measure the actual torque and the observed torque determined with the torque deflection characteristics.

5.1 Shoulder validation

The torque control implemented for the shoulder joint (q2) is validated with the set-up presented in figure 5.1. The shoulder is fixed on a beam such that the arm is not possible to move. With this set-up gravity does not influence the measurements since the arm is straight-up and fixed to the beam. The force/torque sensor is placed between the robotic arm and the wooden blocks that make contact with the beam.

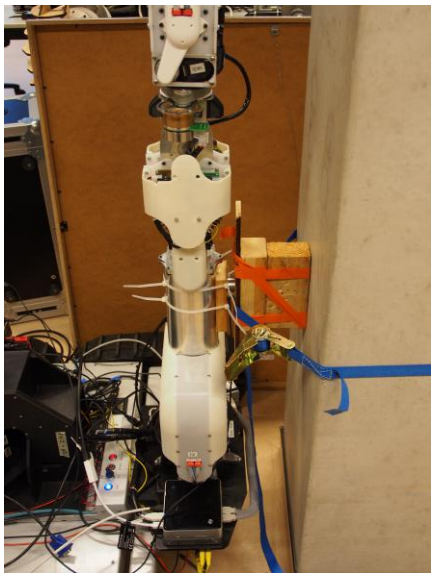


Figure 5.1: Validation set-up shoulder joint(q2).

For 13 pivot positions measurements are performed. Each pivot positions is validated with 7 torque trajectory set-points, resulting in 101 measurements. To present all results is beyond the scope of this chapter. Instead the results are presented for high, medium and low stiffness of the joint. The results for the joint(q2) are depicted in the figures 5.2 till 5.13.

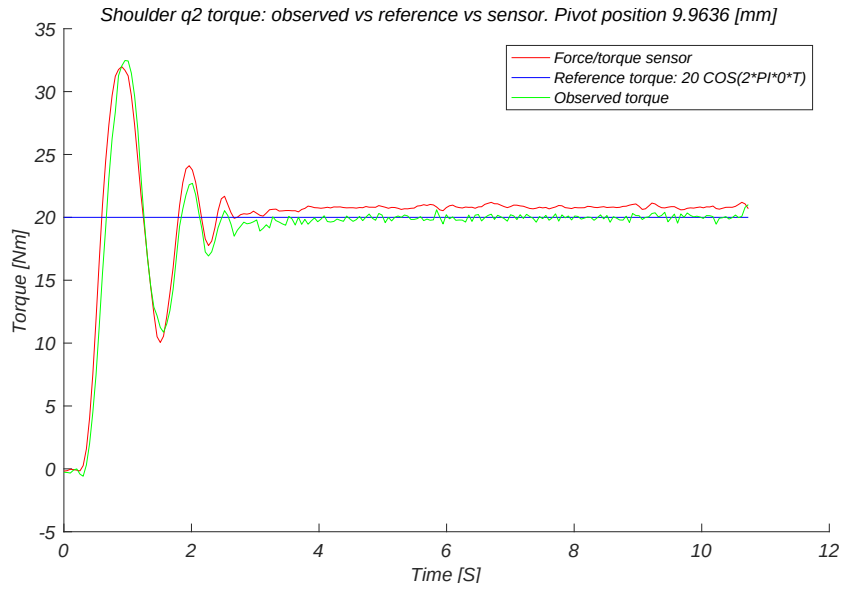


Figure 5.2: Validation of the shoulder (q2) with a step function of 20 [Nm] for a high stiffness setting.

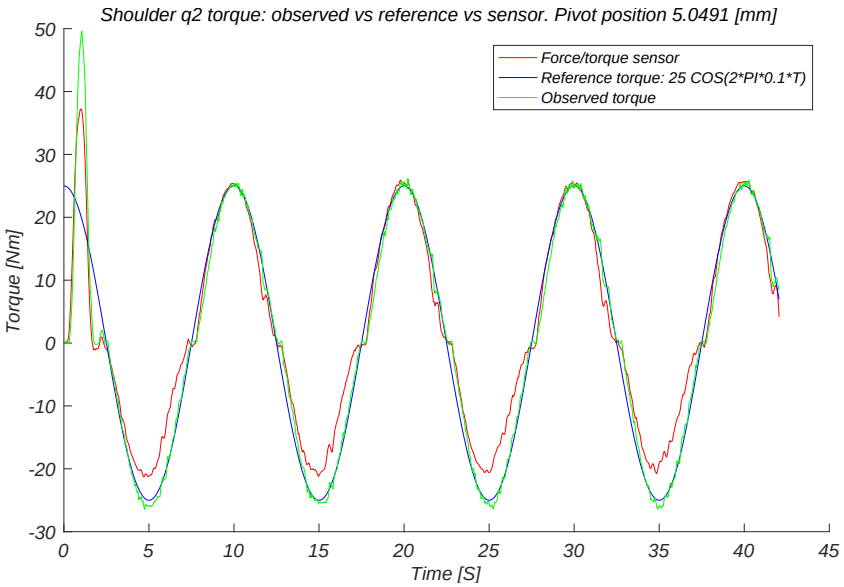


Figure 5.3: Validation of the shoulder (q2) with a cosine function of 0.1 [Hz] for a high stiffness setting.

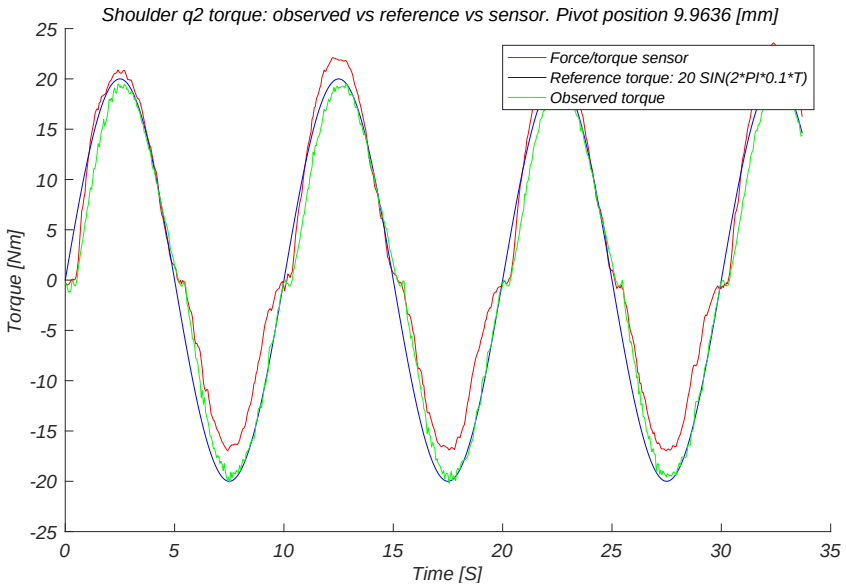


Figure 5.4: Validation of the shoulder (q2) with a sine function of 0.1 [Hz] for a high stiffness setting.

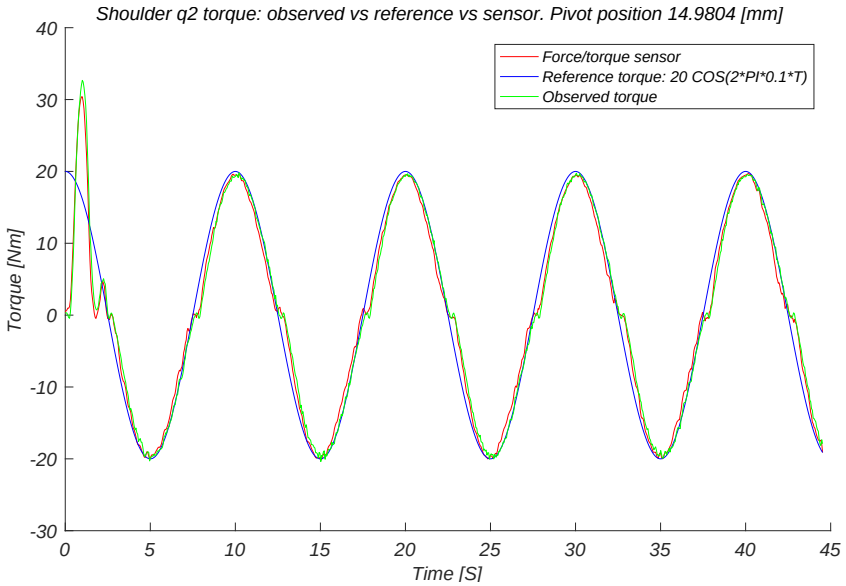


Figure 5.5: Validation of the shoulder (q2) with a cosine function of 0.1 [Hz] for a high stiffness setting

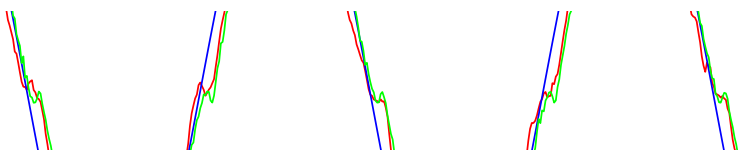


Figure 5.6: Dead-zone in figure 5.5 captured by the observer and force/torque sensor.

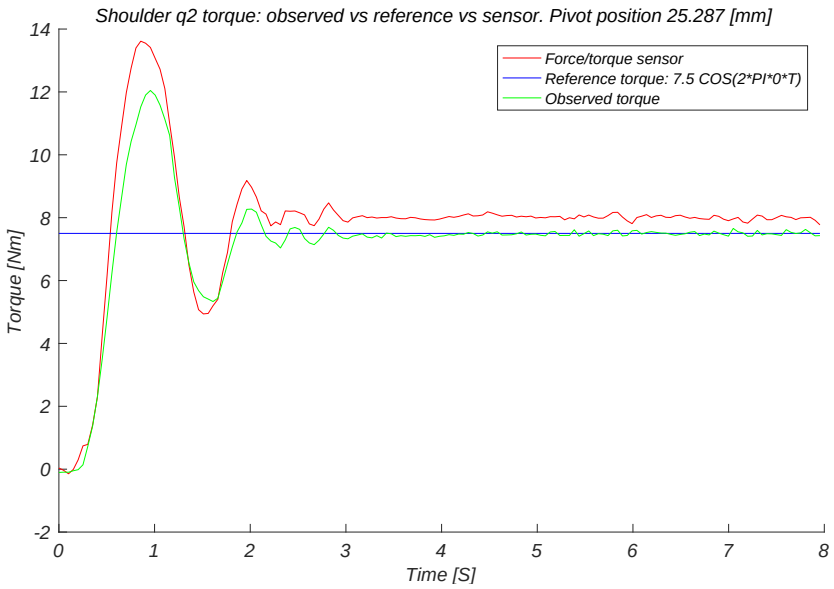


Figure 5.7: Validation of the shoulder (q2) with a step function of 7.5 [Nm] for a medium stiffness setting.

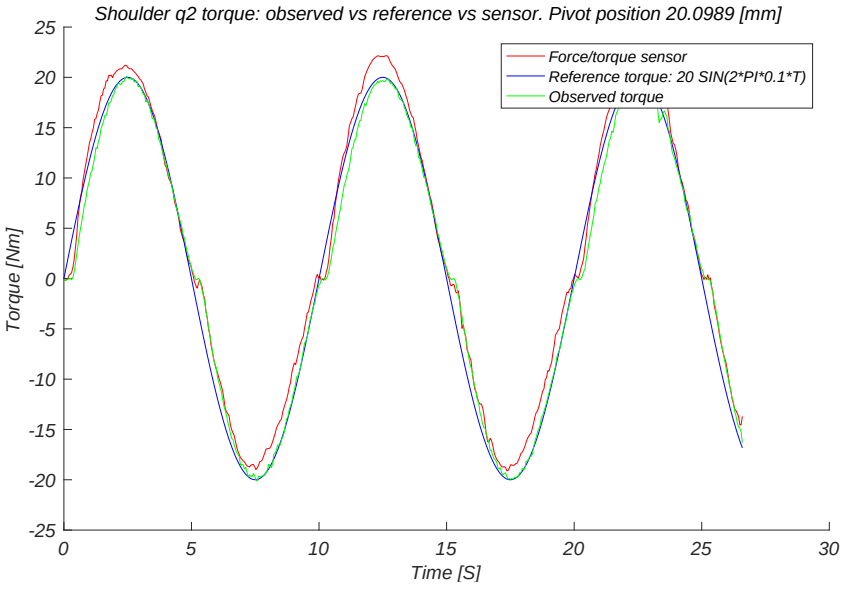


Figure 5.8: Validation of the shoulder (q2) with a sine function of 0.1 [Hz] for a medium stiffness setting.

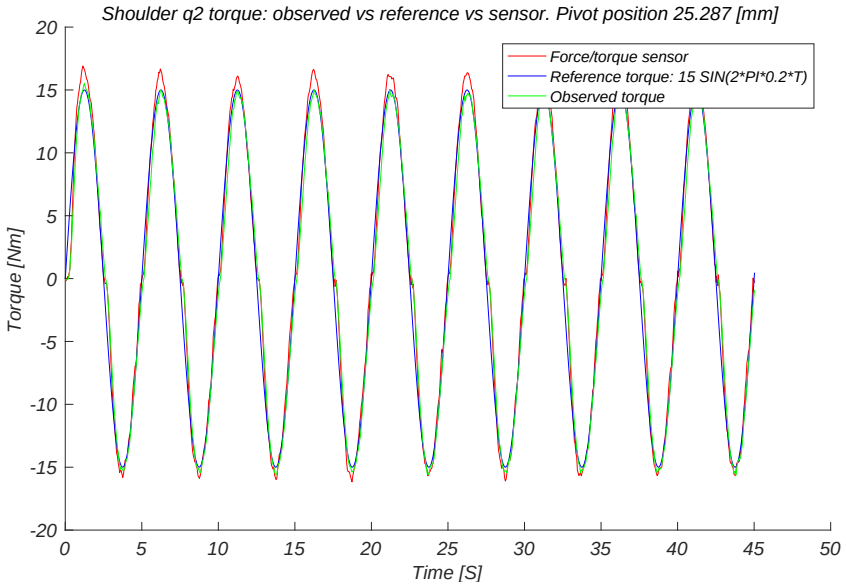


Figure 5.9: Validation of the shoulder (q2) with a sine function of 0.2 [Hz] for a medium stiffness setting.

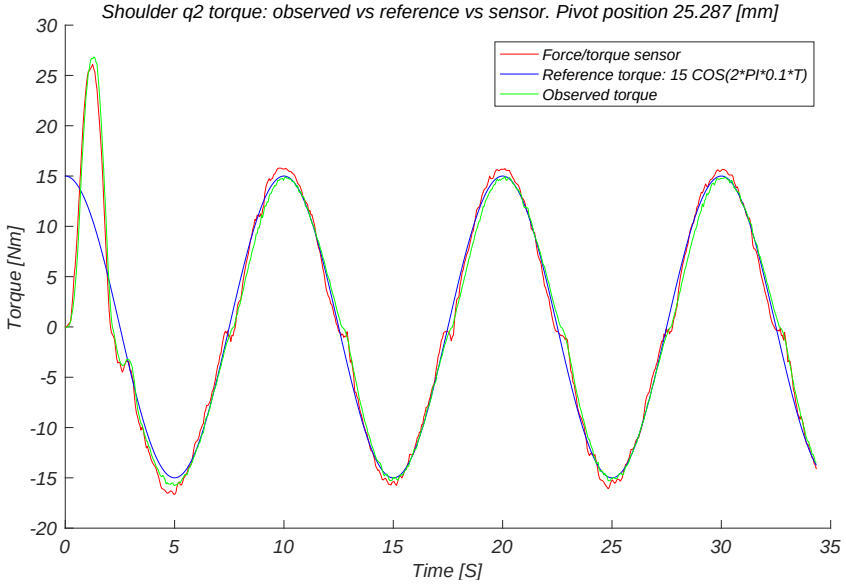


Figure 5.10: Validation of the shoulder (q2) with a cosine function of 0.1 [Hz] for a medium stiffness setting.

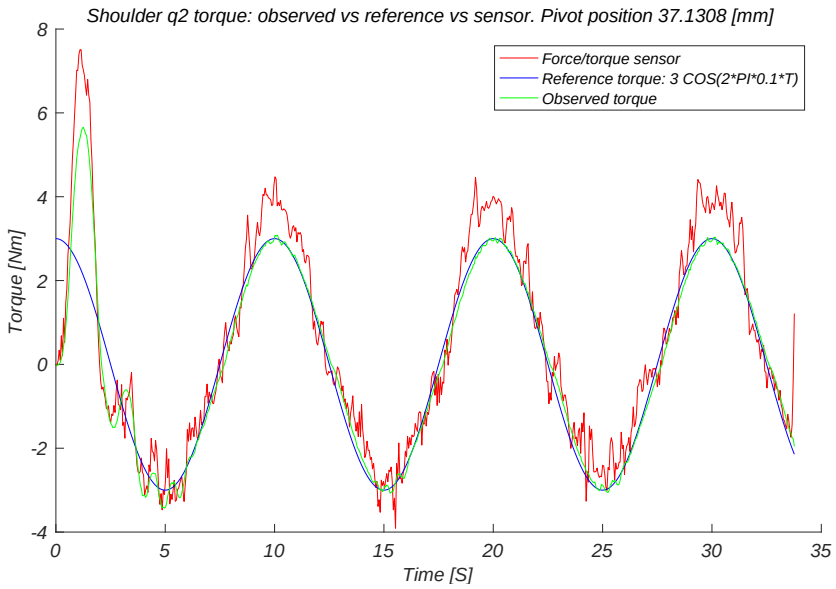


Figure 5.11: Validation of the shoulder (q2) with a cosine function of 0.1 [Hz] for a low stiffness setting.

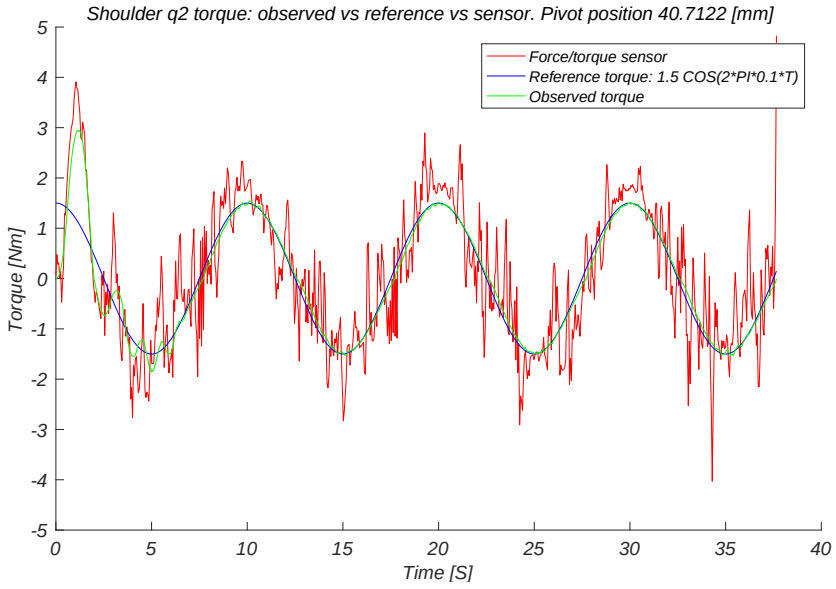


Figure 5.12: Validation of the shoulder (q2) with a cosine function of 0.1 [Hz] for a low stiffness setting.

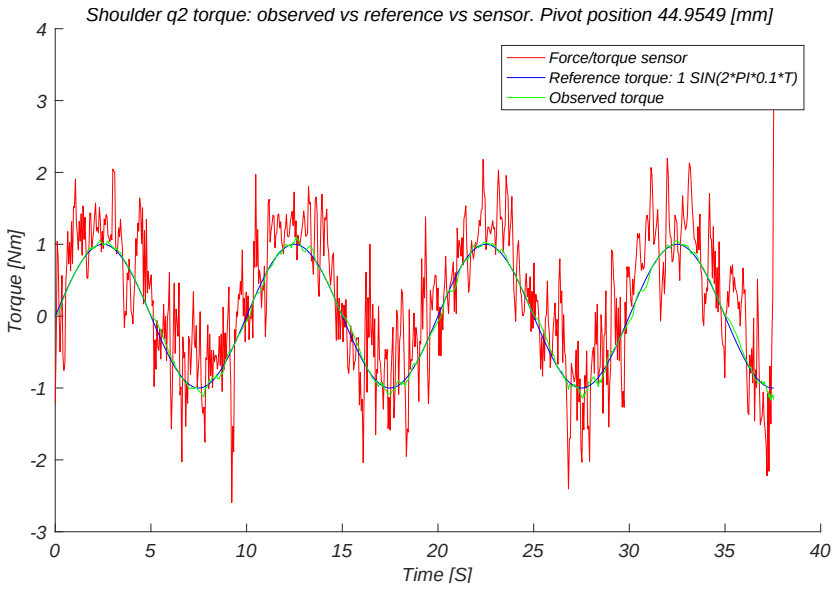


Figure 5.13: Validation of the shoulder (q2) with a sine function of 0.1 [Hz] for a low stiffness setting.

The reference torque is tracked by the observer, which validates the control loop. In some results there is an offset between the observed torque and the measured torque, this is due to an inaccurate fitted torque-deflection curve or the hysteresis and non-linear friction which is not modelled in the current observer. The dead-zone is well captured by the fitted torque deflection curve as depicted in figure 5.6. In figure 5.11 a interpolation between two measured stiffness settings is presented. The observer and measured torque contain similar behaviour and exhibit that interpolation between the characterised stiffness settings is possible. During the low stiffness settings vibrations in the mechanical structure are presents, which introduced noise in the signal of the observer and force/torque sensor. A moving average filter is applied in order compensate for the noise and to be able to see the results. Overall the torque control for the shoulder joint (q2) provides a good tracking of the reference torque and is validated with the force/torque sensor. Improvements are possible by adding the hysteresis and non-linear friction the observer and gain scheduling for the PI control loop to improve the tracking of the observed torque with the reference torque for higher frequencies.

5.2 Wrist validation

The torque control on the 2 DOF wrist joint (q6 and q7) is validated with the set-up presented in figure 5.14. The shoulder joint and the elbow joint are set into a position such that the wrist is pointing straight down towards the floor (z-direction points towards the floor). A mounting plate is constructed such that the force/torque sensor is placed on top of the wrist joint, replacing the gripper. A plate in which a bolt is fed through a hole is is connected with the sensor and is connected to the force/torque sensor. The plate is clamped in a vice tool and is fixed to the ground floor. Which results that the bolt connected to the sensors is free to rotate and translate in z-directions. All other translations and rotations are constrained.

For 14 pivot positions measurements are performed. Each pivot positions is validated with 14 torque trajectory set-points, resulting in 196 measurements. The results for the 2 DOF wrist joint (q6 and q7) are depicted in the figures 5.15 till 5.26

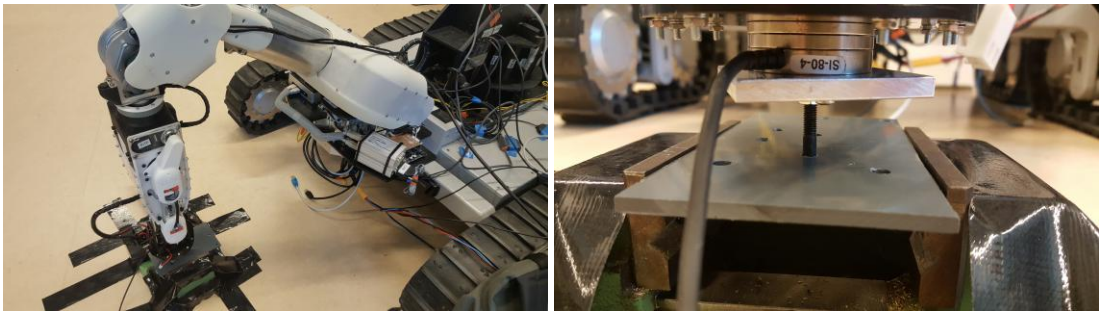


Figure 5.14: Validation set-up wrist joint(q6 and q7).

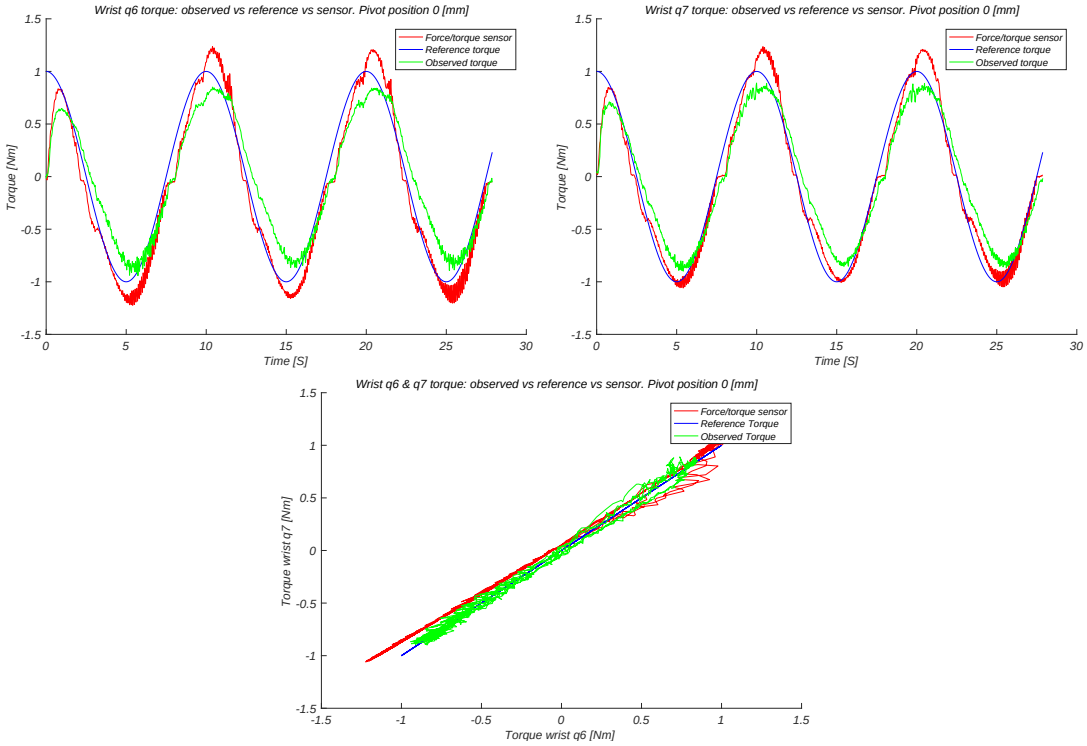


Figure 5.15: Validation of the wrist (q6 and q7) with two cosine functions of 0.1 [Hz] for a high stiffness setting.

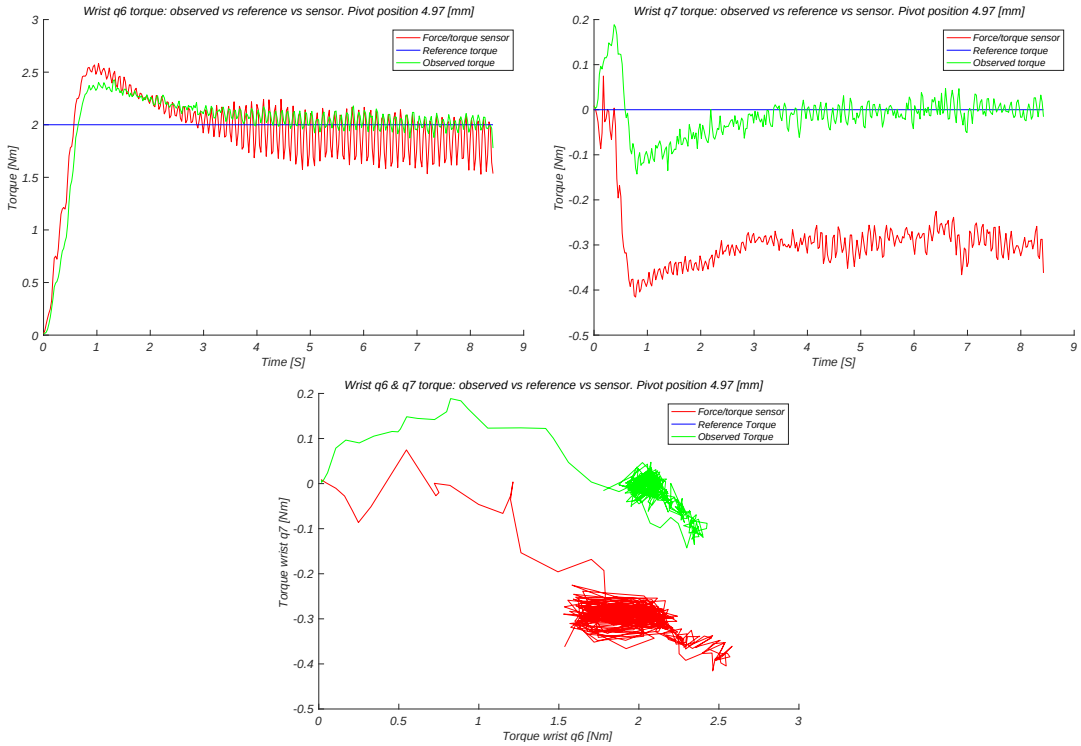


Figure 5.16: Validation of the wrist (q6 and q7) with a step function (q6) of 2 [Nm] for a high stiffness setting.

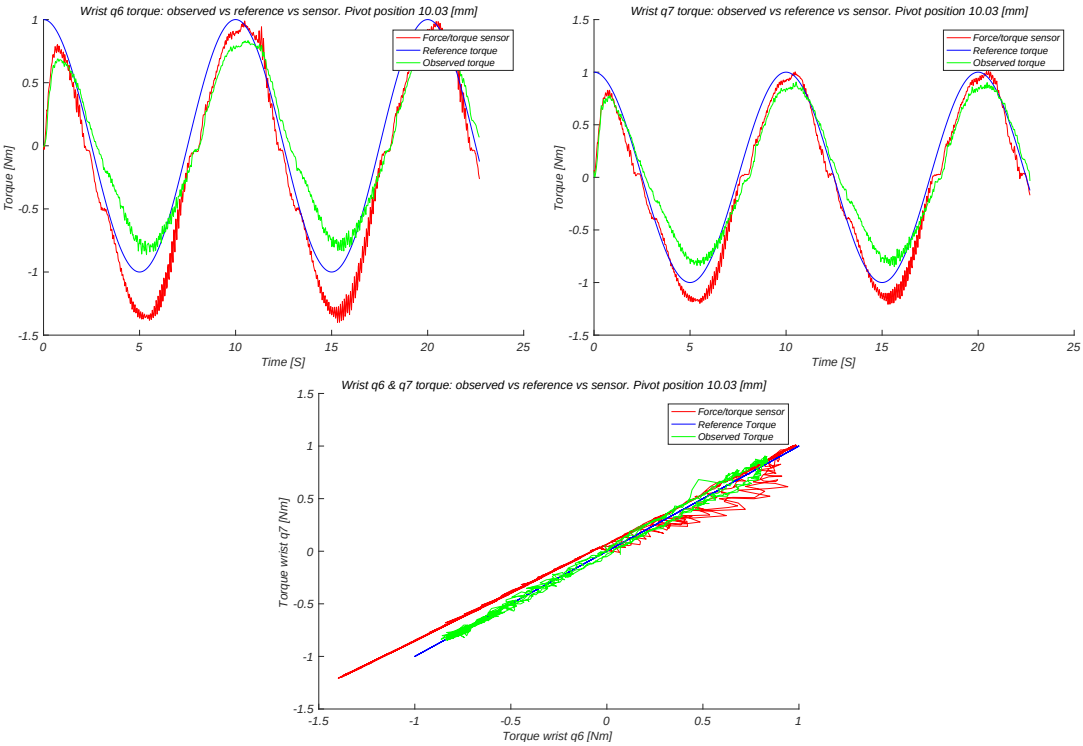


Figure 5.17: Validation of the wrist (q6 and q7) with two sine functions of 0.1 [Hz] for a high stiffness setting.

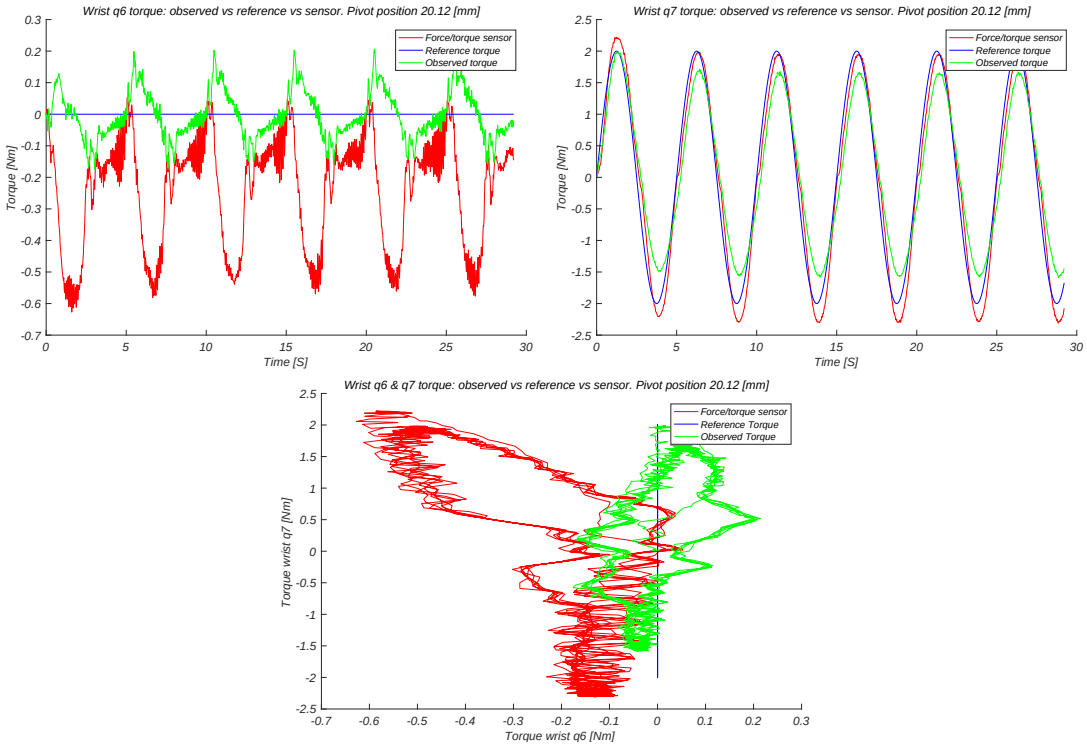


Figure 5.18: Validation of the wrist (q6 and q7) with a sine function (q7) of 0.2 [Hz] for a high stiffness setting.

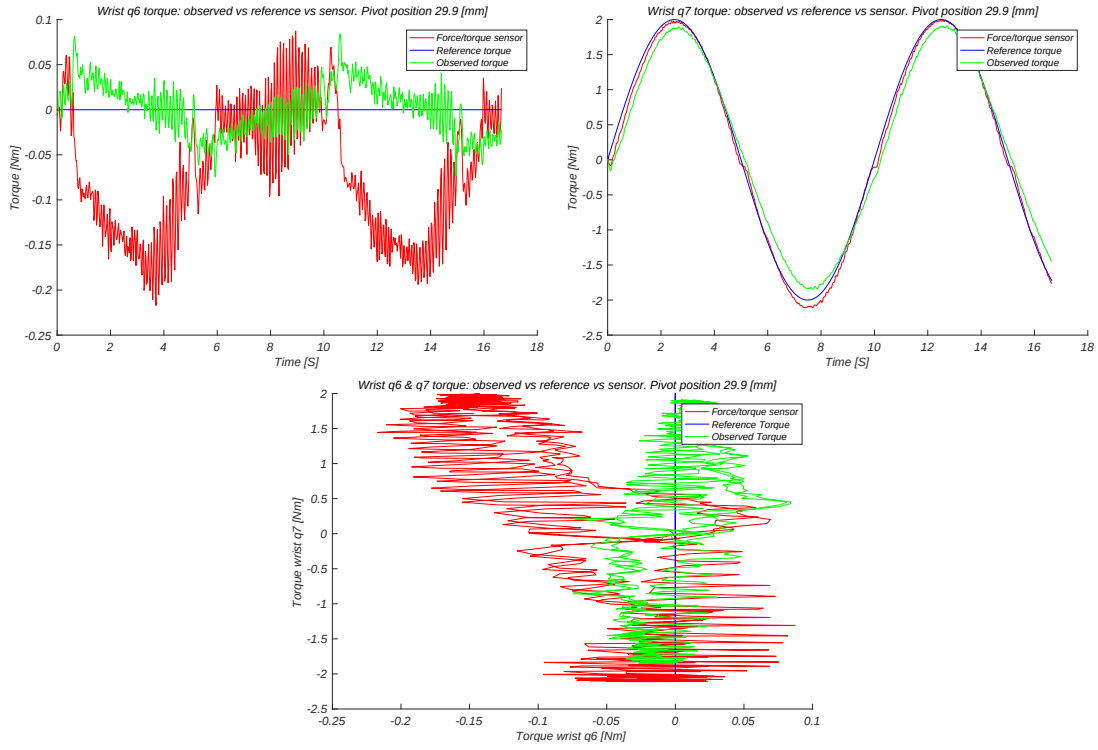


Figure 5.19: Validation of the wrist (q6 and q7) with a sine function (q7) of 0.1 [Hz] for a medium stiffness setting.

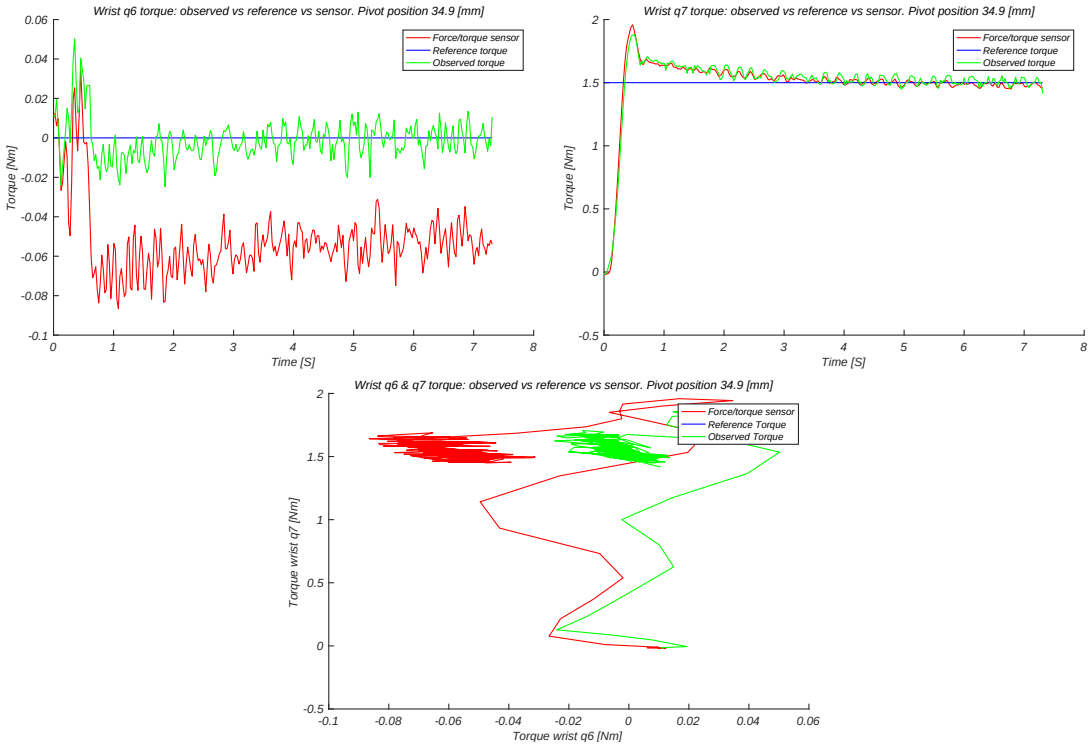


Figure 5.20: Validation of the wrist (q6 and q7) with a step function (q7) of 1.5 [Nm] for a high medium setting.

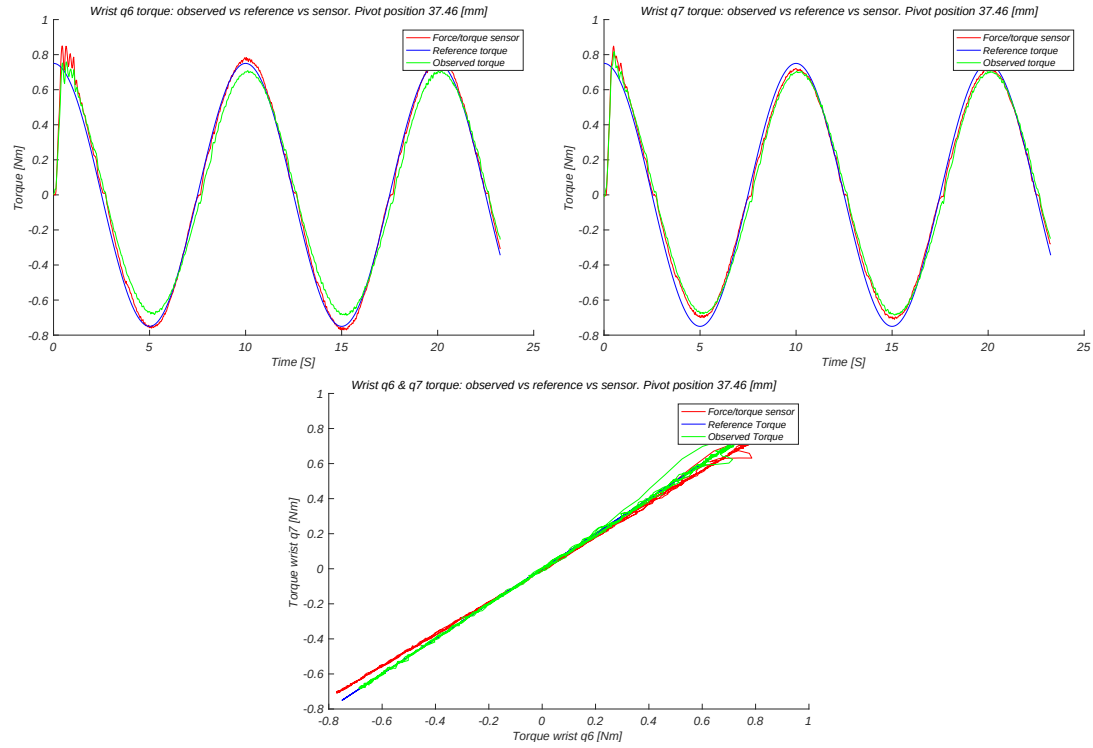


Figure 5.21: Validation of the wrist (q6 and q7) with two cosine functions of 0.1 [Hz] for a medium stiffness setting.

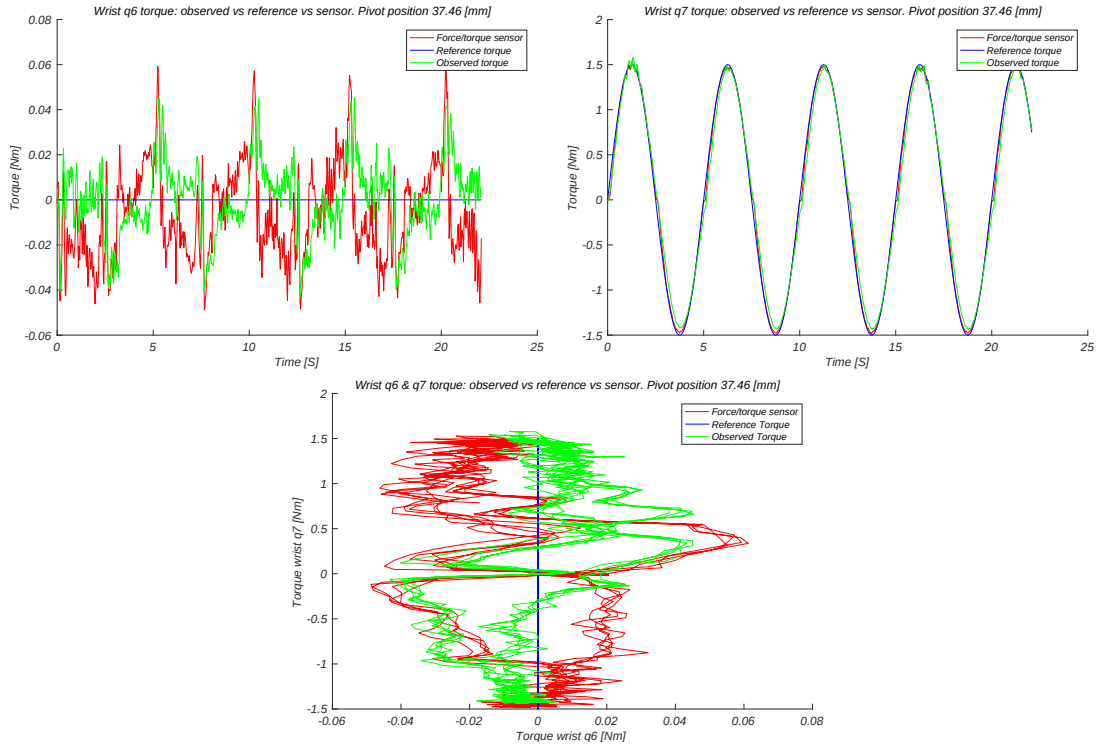


Figure 5.22: Validation of the wrist (q6 and q7) with a sine function (q7) of 0.2 [Hz] for a medium stiffness setting.

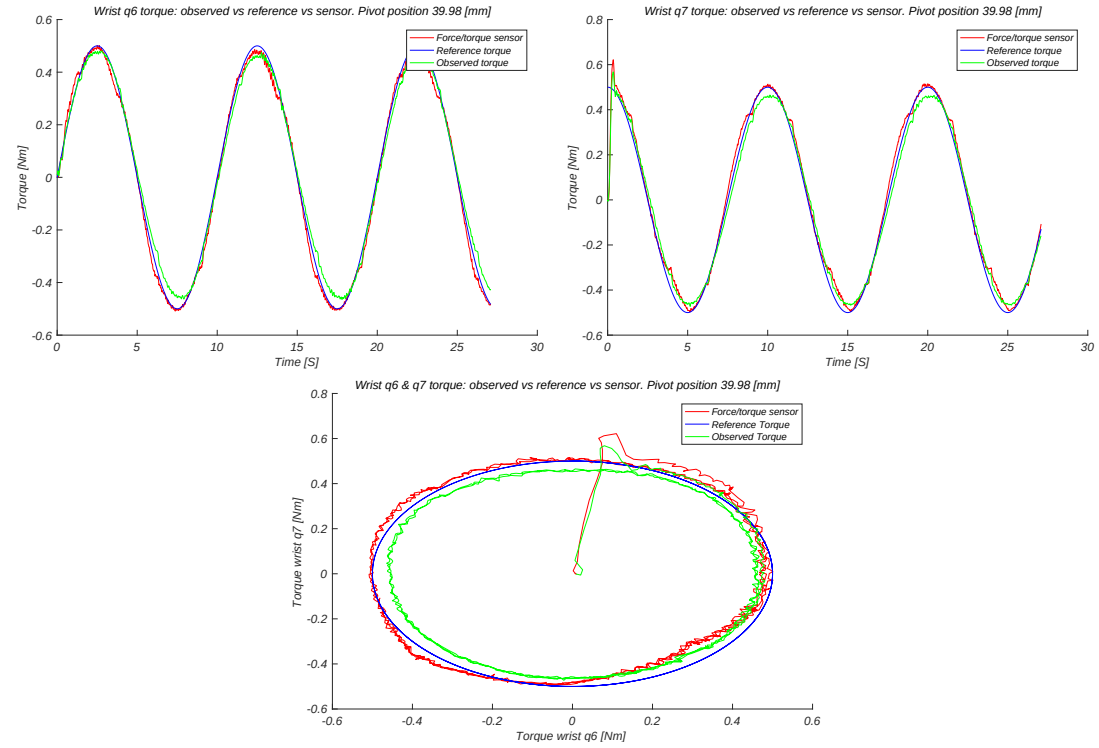


Figure 5.23: Validation of the wrist (q6 and q7) with a sine (q6) and cosine (q7) function of 0.1 [Hz] for a low stiffness setting.

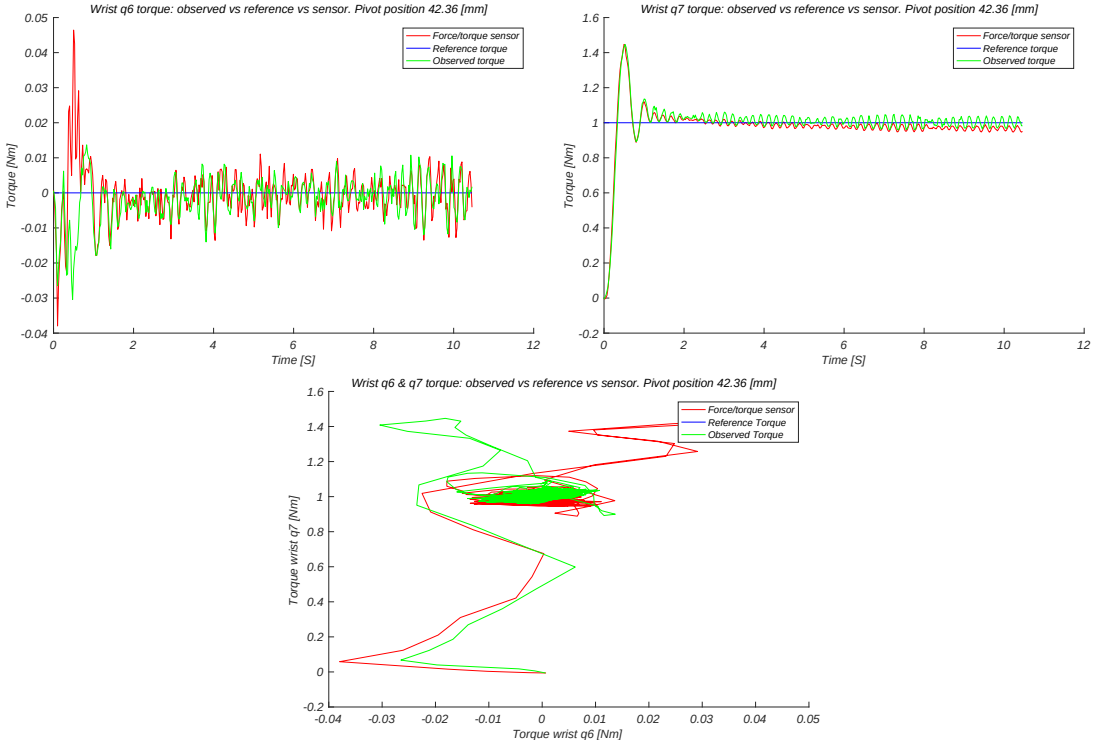


Figure 5.24: Validation of the wrist (q6 and q7) with a step function (q7) of 1 [Nm] for a low stiffness setting.

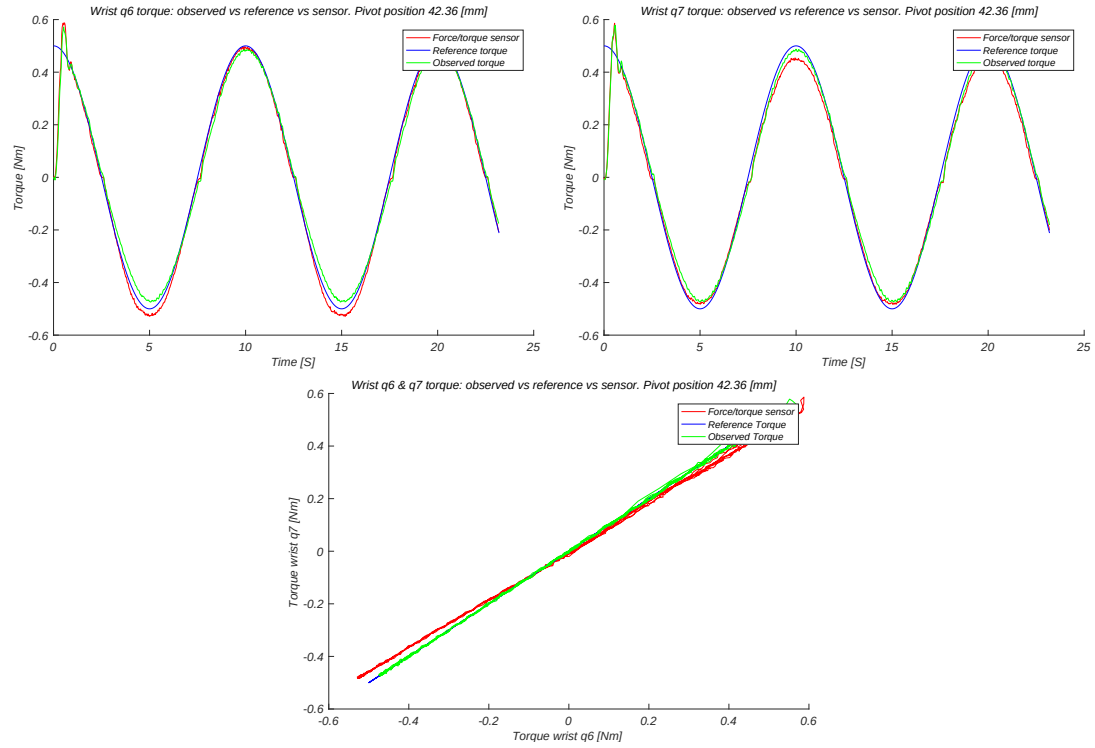


Figure 5.25: Validation of the wrist (q6 and q7) with two cosine functions of 0.1 [Hz] for a low stiffness setting.

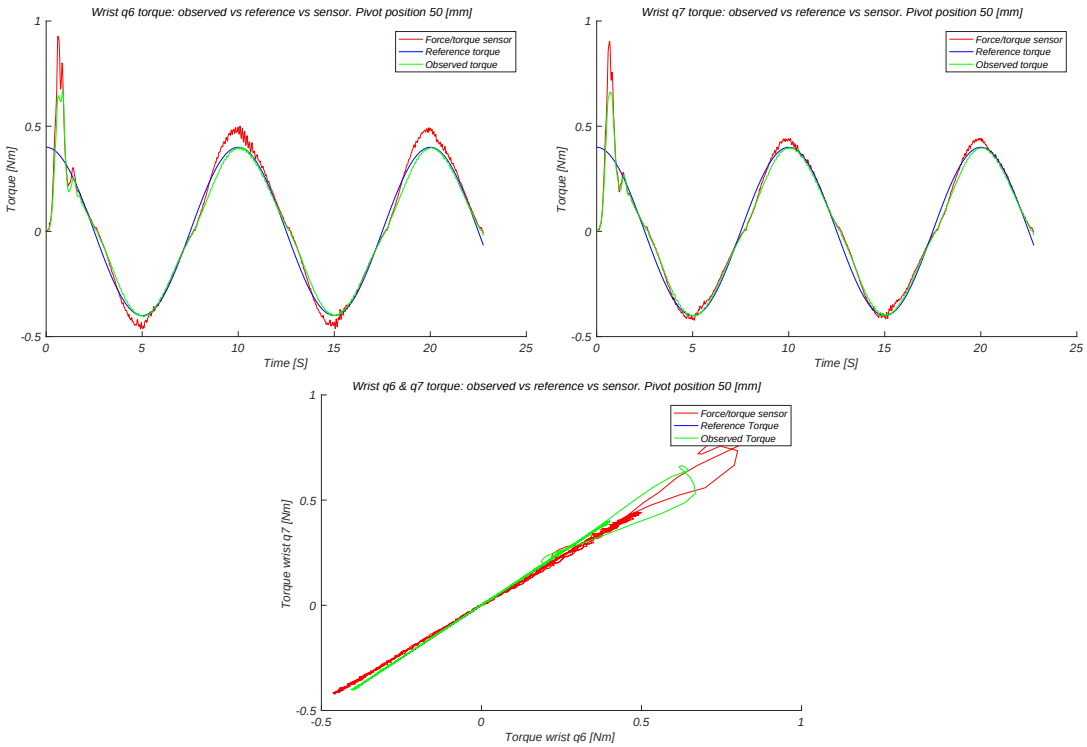


Figure 5.26: Validation of the wrist (q6 and q7) with two cosine functions of 0.1 [Hz] for a low stiffness setting.

The reference torque for a high and medium stiffness setting is not optimal tracked by the observer, which suggest that the PI control loop should be tuned to be able to track the reference torque. The reference torque for a low stiffness setting is tracked by the observer, which validates the control loop. During the high stiffness settings vibrations in the mechanical structure are presents, which introduced noise in the signal of the observer and force/torque sensor. The linear interpolation between the characterised stiffness setting is validated in 5.21, 5.22, 5.24 and 5.25. The observer tracks the reference torque and the force/torque sensor validates the correct estimated torque obtained by linear interpolation of different torque deflection curves in the observer. In some results there is an offset between the observed torque and the measured torque, this is due to an inaccurate fitted torque-deflection curve, linear interpolation between the measured torque-deflection curves or the hysteresis and non-linear friction which is not modelled in the current observer. Overall the torque control for the wrist joints (q6 and q7) provides a good tracking of the reference torque and is validated with the force/torque sensor. Improvements are possible by adding the hysteresis and non-linear friction in the observer and gain scheduling for the PI control loop to improve the tracking of the observed torque with the reference torque for all stiffness settings and higher frequencies. For more results of the validation of the wrist joints (q6 and q7), see appendix D.

6 Conclusions and Recommendations

6.1 Conclusions

This thesis shows that joint level torque control can be achieved on the SHERPA arm with identification and modelling of the joint characteristics. This achieved by regulating the deflection of the compliant joints. For each joint a dynamic model in bond-graphs is created and implemented in 20-sim. The dynamic models represent the behaviour of the joints, however improvements can be made by adding the non-linear friction models in the bond-graphs. The joints are characterised and curve fitting is applied to obtain the torque-deflection characteristics.

The acquired torque-deflection characteristics are implemented in the high-level software. For torque control the observer is extended with the deflection, observed torque and stiffness for each joint. A control loop with a PI controller is written to perform torque control. The graphical user interface enables the user to set a desired torque trajectory and to start torque control. The joint controllers convert each desired joint position into motor positions. These set-points are sent to the ELMO's using the CAN bus. These will perform local position control to achieve the given set-point.

Overall the torque control for the shoulder (q2) and the wrist joints (q6 and q7) provides a good tracking of the reference torque and is validated with experiments. In some results there is an offset between the observed torque and the measured torque, this is due to an inaccurately fitted torque-deflection curve, linear interpolation between the measured torque-deflection curves or the hysteresis and non-linear friction which is not modelled in the current observer.

With this approach joint level torque control for the variable stiffness joints is accomplished and the objectives for this master thesis are satisfied.

6.2 Recommendations and future work

In order to improve the performance of the torque control and for future work the following recommendations are suggested.

- Extend the dynamic models and observer with the hysteresis and non-linear friction models. This improves simulations and the estimated torque of the observer.
- Create a dynamic model of the whole arm, to be able to perform simulations with the complete arm.
- Extend the controller with gain scheduling for the PI control loop to improve the tracking of the observed torque with the reference torque for all stiffness settings and higher frequencies.
- Tune the ELMO motor drives to be able to track higher reference frequencies.
- Implement impedance control on the SHERPA arm.

A Can Interface

In the SHERPA high level software the communication with the ELMO servo controllers is accomplished by a CAN-bus. It is possible to sent motor set-points and to obtain the data of the motor encoders. Initially some of the motor encoders are used to determine the deflection of a joint. However the CAN-bus is not reliable and randomly messages are lost. In that case the received encoder values are zero, this results in spikes in the observed deflection and position of each joint. Leading to undesired behaviour of the arm. Occasionally the arm falls a few centimetres and would then return to its correct position. The torque control loop relies on the observed deflection. The observed torque is estimated with the deflection. If the deflection contains spikes, the observed torque contains spikes. Which results that the control loop becomes unstable or the given torque trajectory set-points are not tracked. An example is presented in figure A.1. In this figure the observed torque of the shoulder (q2) is presented, together with the required torque trajectory (Reference) and the actual measured torque with the force/torque sensor.

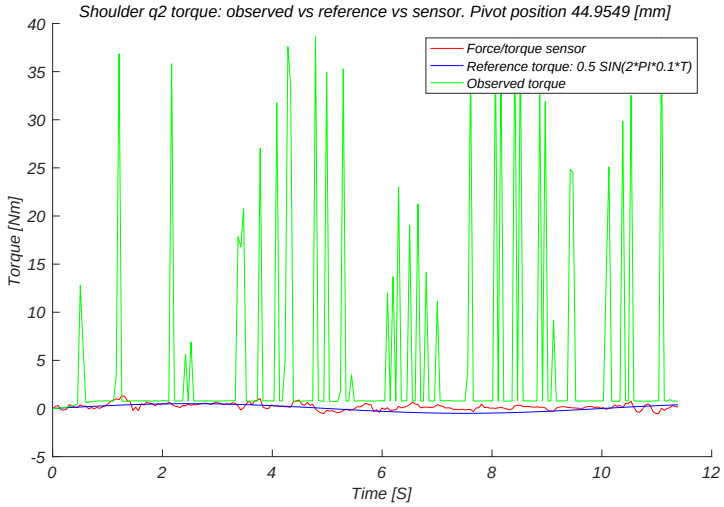


Figure A.1: Observed deflection in the shoulder (q2) with spikes due to lost CAN-bus messages.

In (Boterenbrood, 2015) a design and implementation of the CAN-bus is accomplished. In this project, a low-level software interface is developed that is able to drive the arm from ROS. A ROS software node is presented that can drive the motors in the arm over a CAN bus. The ROS software node supports driving motors on multiple CAN buses. The software architecture is presented in A.2.

The ROS-CAN Interface contains a part to send frames and to receive CAN frames. These are the CANSender and CANReceiver blocks, both with their own queue to prevent loosing messages. To support multiple CAN device types a separate control block for each device type can be added. The incoming CAN frames are send to the right device specific control block by the NodeContainer. The NodeContainer is responsible for creating the instances of the device specific control block and keeping a list that maps each CAN node id to the corresponding device specific control block. The CANSender sends all frames from the SendQueue to the CAN bus, before sending a frame to the CAN bus it will send it to the SendFilterQueue of the CANReceiver. When the CANReceiver receives a frame it will compare this with the SendFilterQueue and if a match is found it will drop the frame and remove it from the SendFilterQueue, if no match is found the frame is added to the ReceiveQueue. The CANController will process all frames from the ReceiveQueue.

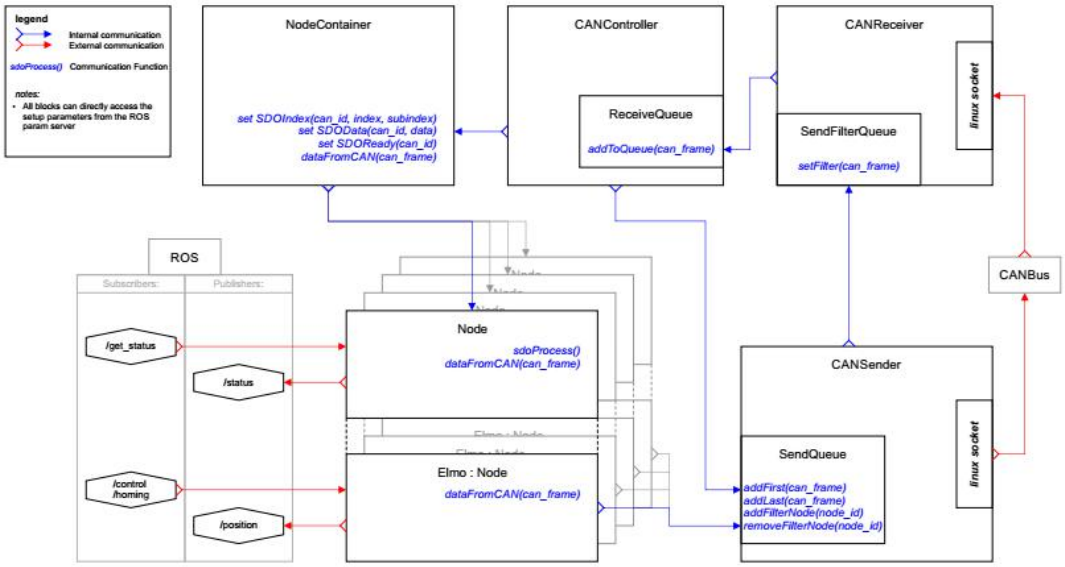


Figure A.2: ROS-CAN interface diagram (Boterenbrood, 2015).

To resolve the problem of the lost messages, the implementation of the CAN-bus in the high-level software was investigated. Filters were implemented for the received messages, however the result was not enough and due to limited time it was not possible to resolve the problem. It is possible that the data is lost due to the wiring of the CAN-bus, at this moment this wiring unshielded. Resulting in that external magnetic fields could influence the data. It is therefore necessary to only use the absolute encoders to determine the position or deflection of a joint.

B Surface fit torque deflection characteristics of the wrist

A other approach that was examined is to fit the torque deflection characteristics of the wrist joint on a 3D surface. A surface is fitted on the 4 fitted 4th order polynomial curves for each stiffness setting. This results in 11 surfaces, only one stiffness setting will be elaborated in this appendix. Between the surfaces and thus the measured pivot positions, linear interpolation is used. This method has the advantages that it is simple to implement in the high level software. The fitted surface is a third order polynomial surface, which is described as B.1. Higher or lower order polynomial surface fits provided a less accurate result. In order to fit the surface on the four fitted curves for the measured angles 0,45,90 and 135 degrees, the curves are converted to polar coordinates and the absolute value of the measured torque is taken. The relationship is described in B.2, where x and y represent the deflection of wrist joint q6 and q7 respectively. To determine θ, the quadrant in which the deflection is present is taking into account. In figure B.1 an example of the fitted surface is presented. The fit for the surface is limited to a torque of 5$[Nm]$, deflection of 0.2$[RAD]$ and is forced to contain zero torque and zero deflection to improve the fit.

$$FittedSurface(x,y) = p00 + p10 * x + p01 * y + p20 * x^2 + p11 * x * y + p02 * y^2 + p30 * x^3 + p21 * x^2 * y + p12 * x * y^2 + p03 * y^3 \tag{B.1}$$

$$r = \sqrt{(x^2 + y^2)} \qquad \theta = \tanh(\frac{y}{x}) \tag{B.2}$$

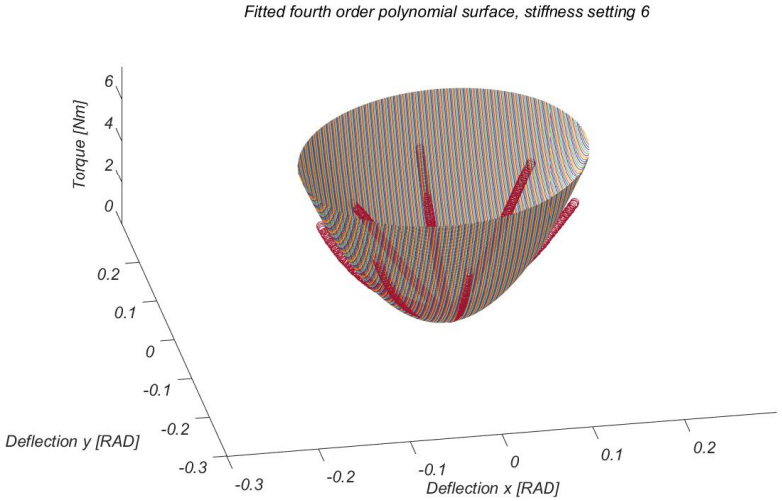


Figure B.1: Fitted Surface for one stiffness setting.

From the fitted surface it is concluded that the fit is not ideal for each curve it is based on. A comparison is made with the initial four curves on which the surfaces is based on, as depicted in figure B.2. Indeed the fit for this stiffness setting is not perfect and could provide an inaccurate torque estimation. The surface fits for each stiffness setting vary in accuracy. For certain stiffness settings the fit is acceptable, while for other stiffness settings the fit is inaccurate.

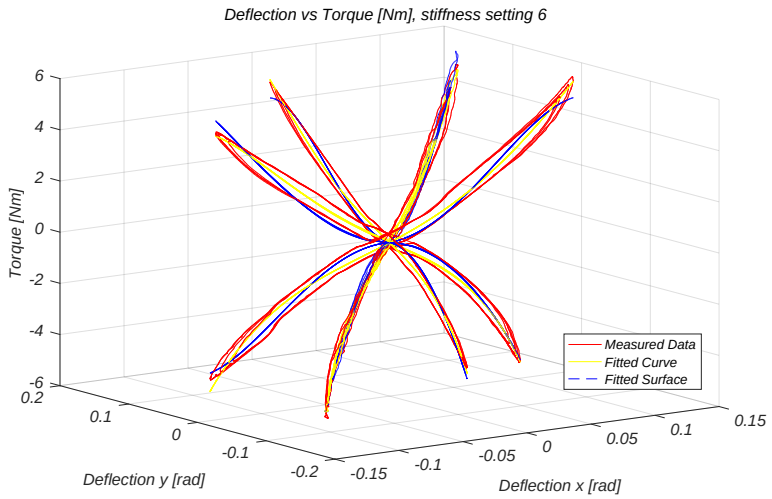


Figure B.2: Fitted surface vs the fitted curves vs the measured data.

Between each stiffness setting, and thus the 11 surfaces, linear interpolation is used. The result is depicted in figure B.3. In this figure the pivot position is changed from 0 mm to 50mm in steps of 7.5 mm. The blue surfaces represent a surface based on a measured pivot position, the green surfaces are interpolated surfaces between two measured pivot positions.

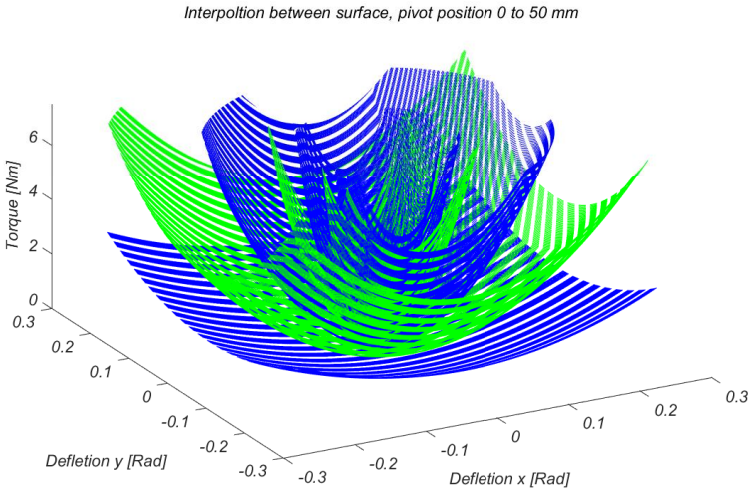


Figure B.3: Interpolation between the surfaces and thus the pivot positions.

These surfaces were implemented in the high level software and measurements were made with the wrist to determine the performance of the estimated torques using surfaces. To determine the torque in each individual joint direction (q6 or q7) the polar coordinates are transformed to Cartesian coordinates, described in B.3.

$$x = r * \cos(\theta) \qquad y = r * \sin(\theta) \tag{B.3}$$

This approach resulted in poor performance, such that the desired torque trajectory is not followed and the estimated torque is not accurate. The surface fit is to inaccurate to represent the measured torque deflection characteristics. Secondly the fitted surface is oscillating in the hysteresis of the measured torque deflection characteristics, this is highly undesired. Since it is impossible to extend the fitted surface with a model to take the hysteresis into account.

C Gravity Compensation

In order to improve the calibration of the arm and to be able to use gravity compensation in future work the Unified Robot Description Format(URDF) in combination with a kinematic solver is used. The URDF is a markup language, designed to describe robots. The description is described in a special XML format. The most important elements in the XML specification are (Barbieri, 2016):

- Link: Describes the properties of a specific robot link. Each link must have a unique name. The visual, inertial and collision details are configured in the corresponding subtags of the link element. The visual part as well as the collision model can either be composed from primitive shapes or from mesh files.
- Joint: Describes the properties of a joint. A joint is a flexible connection between two links, having exactly one parent and one child link. Each joint states a new reference frame for its child link and it is positioned relative to its parent frame. The joint actuates its child link relative to its parent link along the joint axis. There are different types of joints available such as fixed and revolute. A fixed joint states a rigid connection between parent and child link. A revolute joint is a rotational joint with one degree of freedom. Details like angular joint limits, axis orientation and the dynamic properties of the joint motors can be configured in the corresponding subtags of the joint element. Those elements are used to form the URDF graph that exactly describes the kinematic chain of the robot components and their placement relative to each other.

The kinematic solver provided by the MoveIt! package is able to calculate the gravity compensation torques, taking into consideration the pose of the arm and the inertia's, center of masses and the kinematic relations between the joints and links defined in the URDF file. To verify the calculated gravity compensation torques, the arm is set into a fixed pose as depicted in figure C.1. The arm is set in a angle of 45 degrees and all masses of the joints are set to 1 [Kg]. The mass of the camera and gripper are set to 0 [Kg].

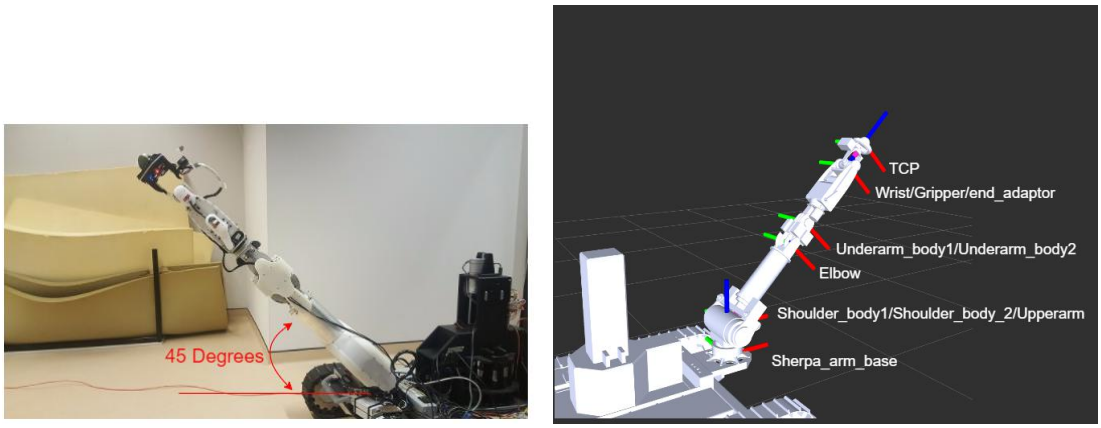


Figure C.1: Experimental set-up to validate the gravity compensation torques.

In the URDF 10 reference frames are defined. The center of masses of each joint is defined in table C.1. In this table all lengths are defined in a positive upward z-direction with respect to it's

own reference frame. The distance between the different reference frames are defined in table C.2.

$Body$	$Center\ of\ mass[m]$
$End\_Adaptor$	0.1
$Wrist$	$Center\ of\ the\ Reference\ Frame$
$Underam\_body1$	0.18
$Underam\_body2$	0.045
$Elbow$	0.025
$Upperarm$	0.33
$Shoulder\_body1$	$Center\ of\ the\ Reference\ Frame$
$Shoulder\_body\_2$	$Center\ of\ the\ Reference\ Frame$
$Sherpa\_arm\_base$	0.0039

(C.1)

$Reference\ frame\ 1$	$Reference\ frame\ 2$	$Distance\ between\ the\ frames\ [m]$
$Wrist$	$Underam\_body1$	0.348
$Underam\_body1$	$Elbow$	0.12
$Elbow$	$Shoulder\_body\_2$	0.45
$Shoulder\_body\_2$	$Sherpa\_arm\_base$	0.1348

(C.2)

Using the geometrical relations each gravity compensation torque is calculated as depicted in equation C.3. A change of sign is required in the calculated torques in order to compensate for the torques. The results of the calculated torques with the kinematic solver and the URDF are presented in figure C.2. The gravity compensation torques calculated with the URDF and kinematic solver are validated. Resulting in that gravity compensation torques can be calculated and used in future work with the URDF and kinematic solver for each pose of the arm.

$$\begin{aligned} Wrist &= 1 * \cos 45^\circ * 9.81 * 0.1 = 0.693[Nm] \\ Upperarm &= 1 * \cos 45^\circ * 9.81 * (0.448 + 0.348 + 0.18 + 0.045) = 7.08[Nm] \\ Elbow &= 1 * \cos 45^\circ * 9.81 * (0.568 + 0.468 + 0.3 + 0.165 + 0.025) = 10.58[Nm] \\ Shoulder &= 1 * \cos 45^\circ * 9.81 * (1.018 + 0.918 + 0.75 + 0.615 + 0.475 + 0.33) = 28.48[Nm] \end{aligned} \tag{C.3}$$

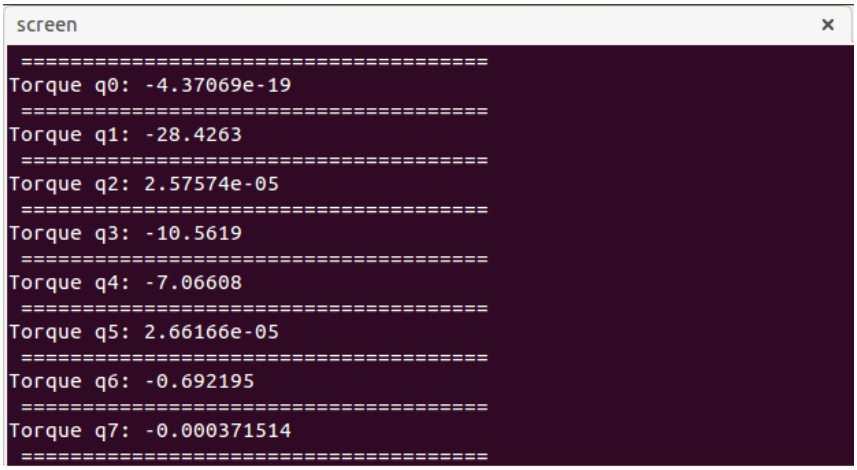


Figure C.2: Calculated gravity compensation torques with the URDF file and kinematic solver.

D Experimental results of the wrist joints

Additional validation results of the wrist joints (q6 and q7) are depicted in the images below.

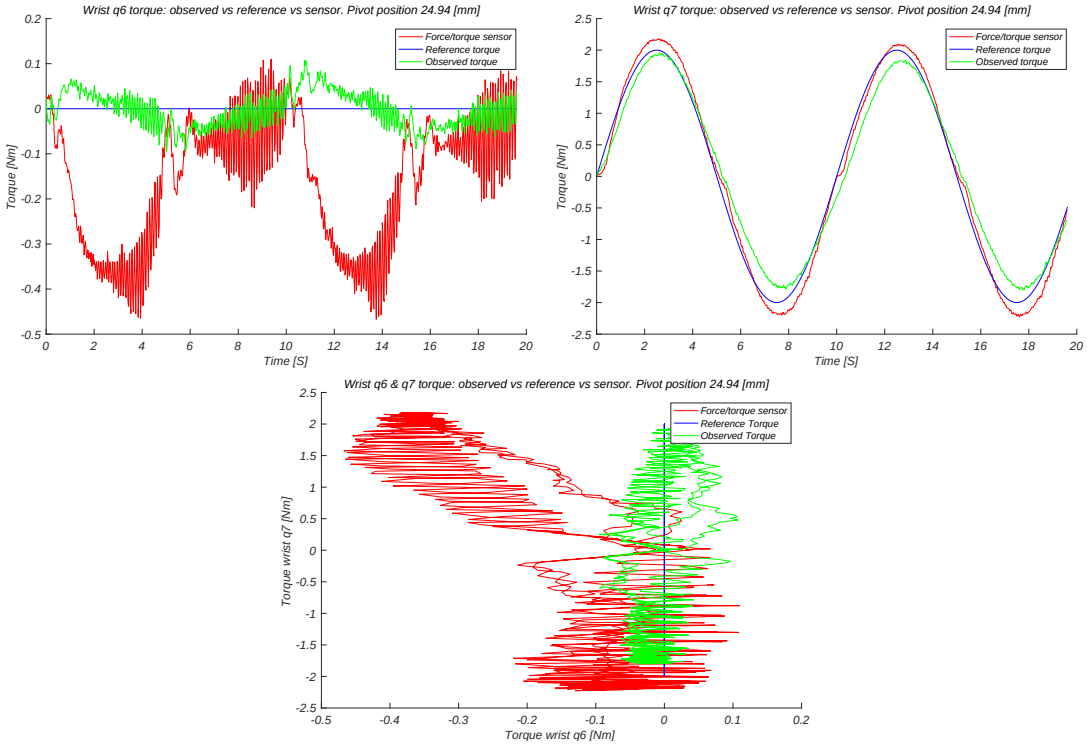


Figure D.1: Validation of the wrist (q6 and q7) with a medium stiffness setting.

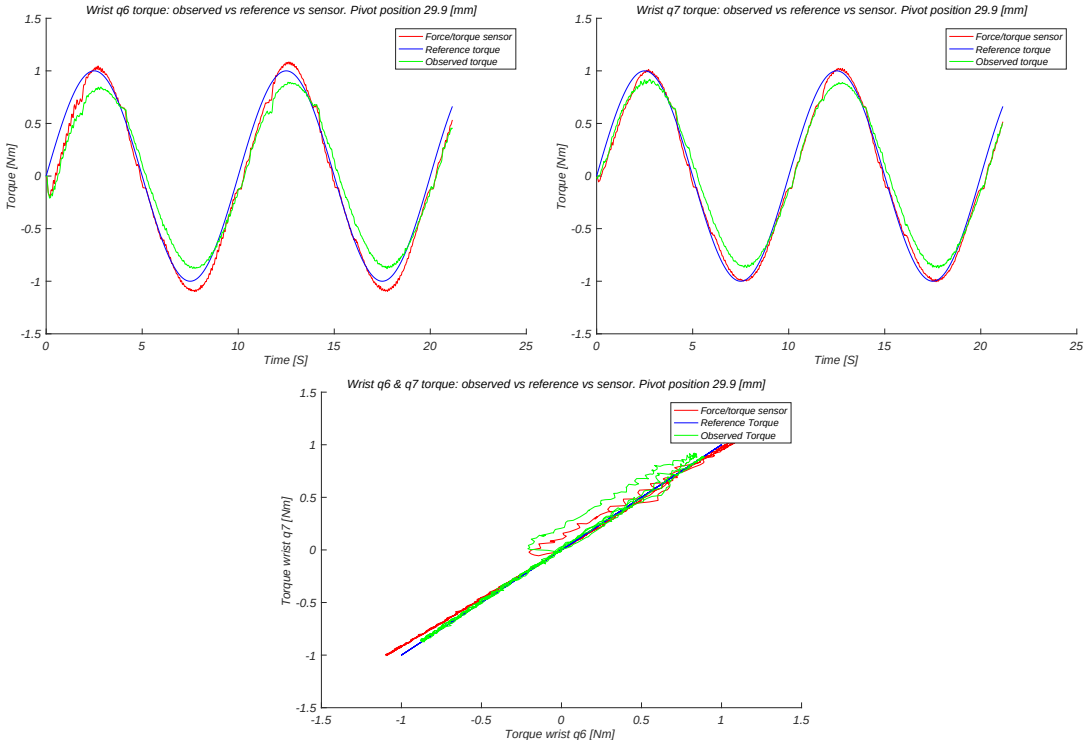


Figure D.2: Validation of the wrist (q6 and q7) with a medium stiffness setting.

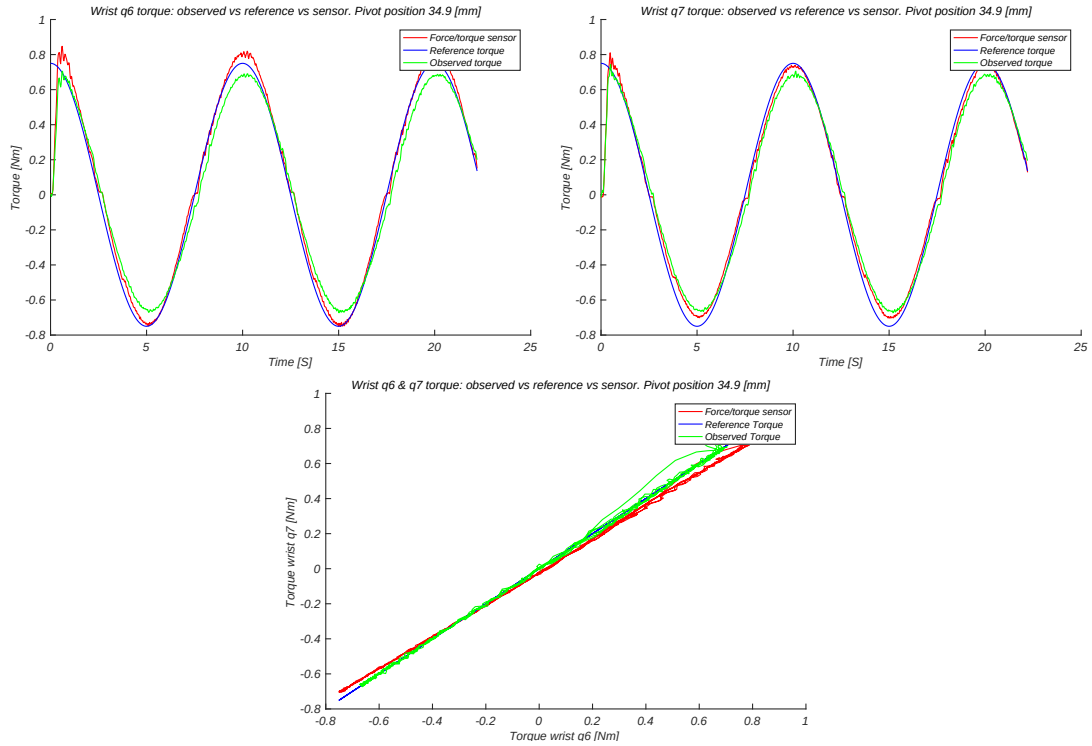


Figure D.3: Validation of the wrist (q6 and q7) with a medium stiffness setting.

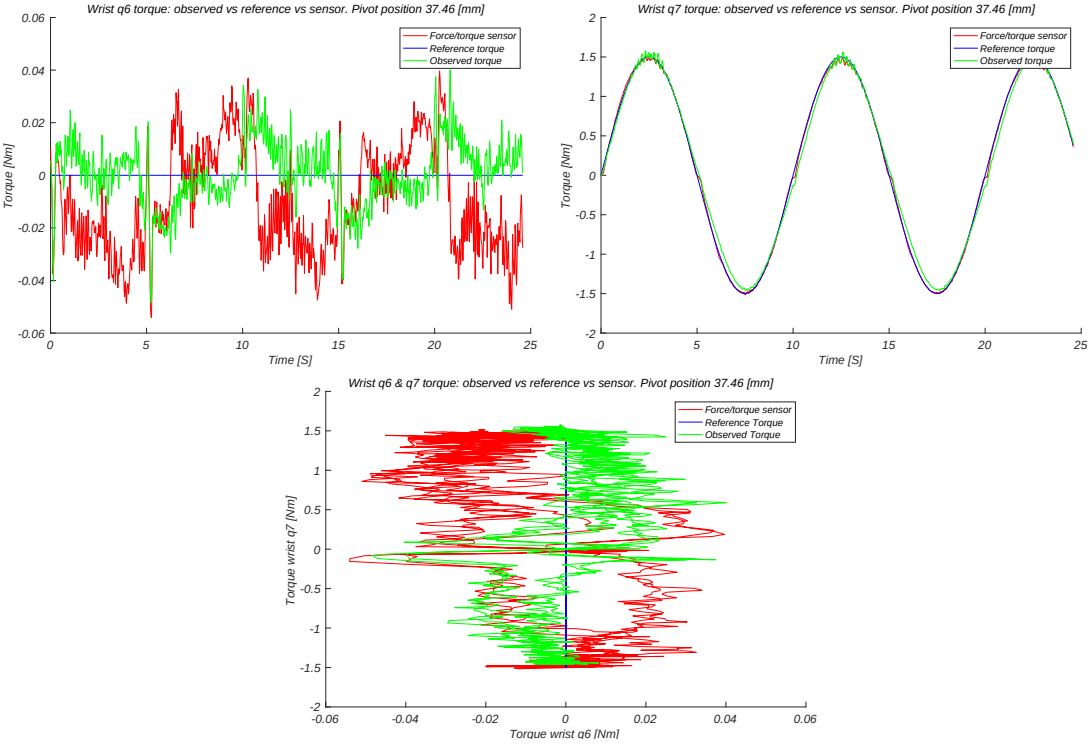


Figure D.4: Validation of the wrist (q6 and q7) with a medium stiffness setting.

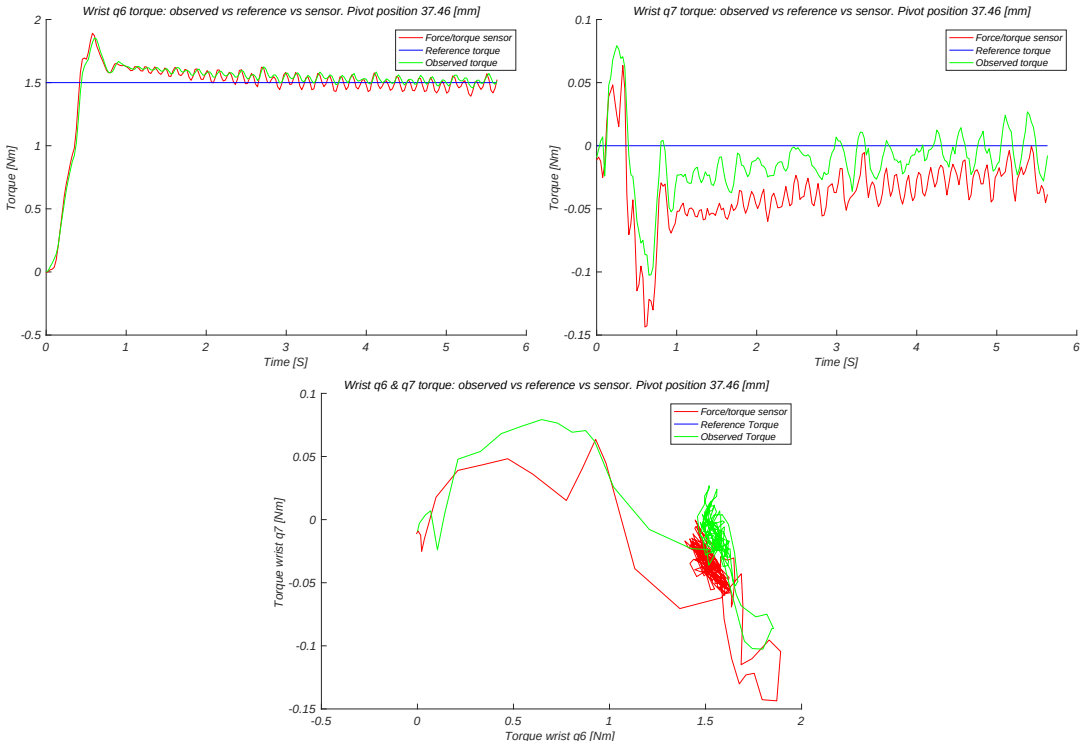


Figure D.5: Validation of the wrist (q6 and q7) with a medium stiffness setting.

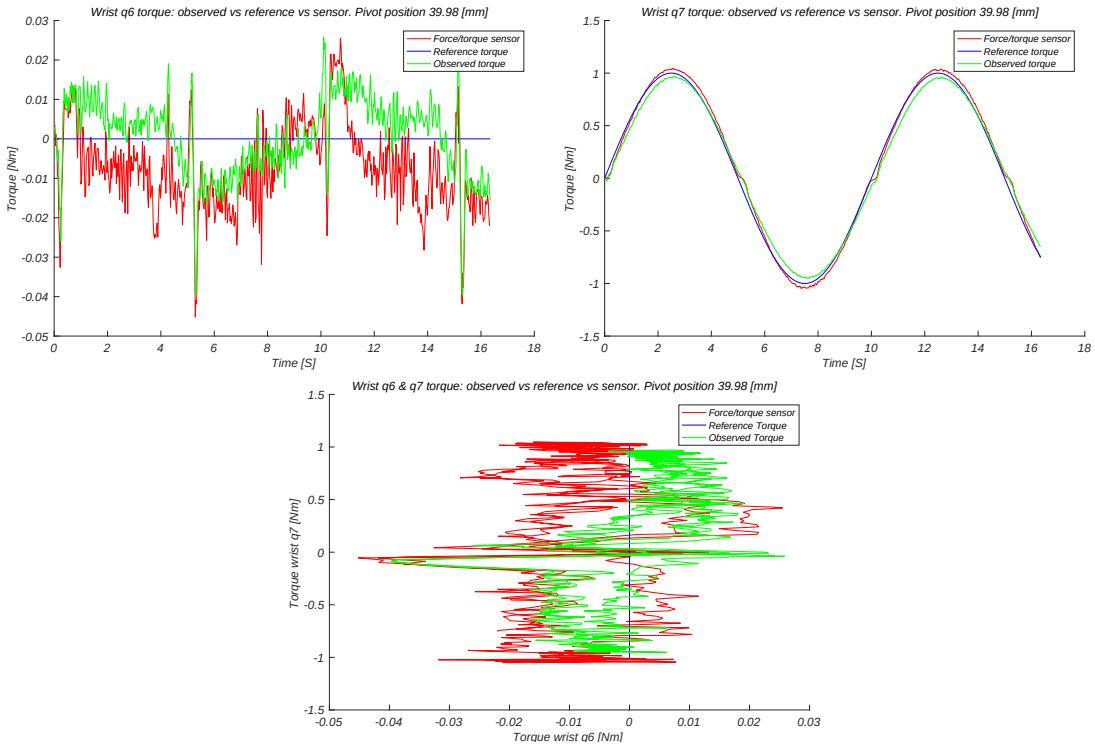


Figure D.6: Validation of the wrist (q6 and q7) with a low stiffness setting.

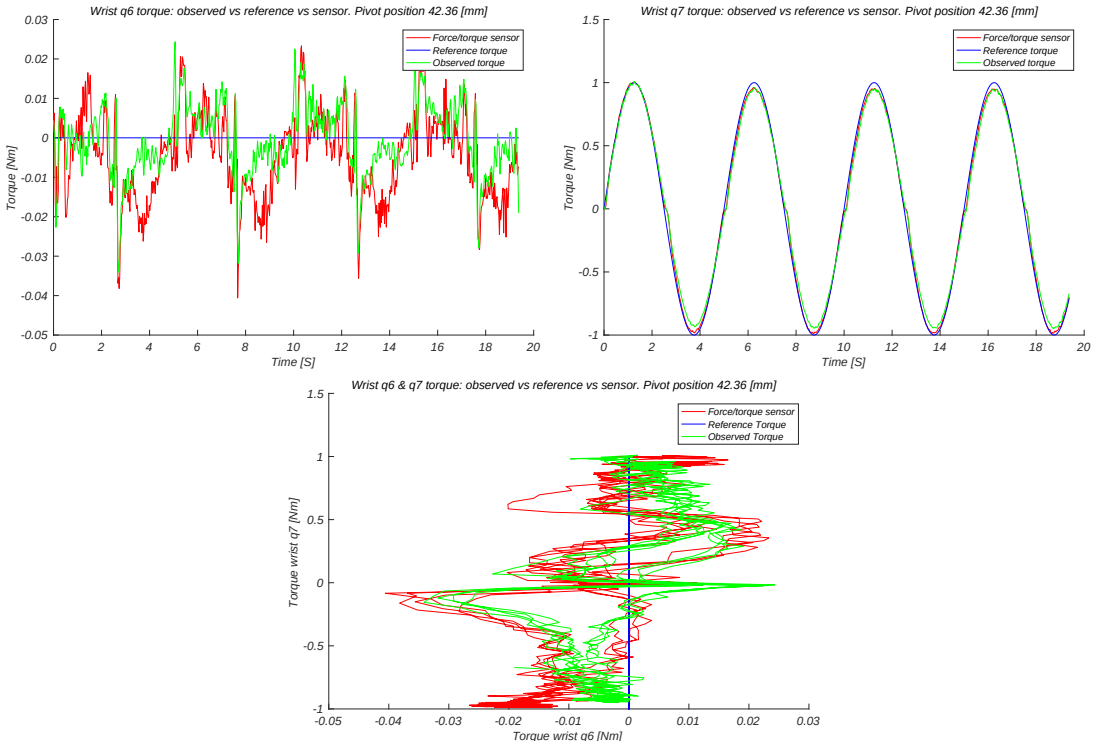


Figure D.7: Validation of the wrist (q6 and q7) with a low stiffness setting.

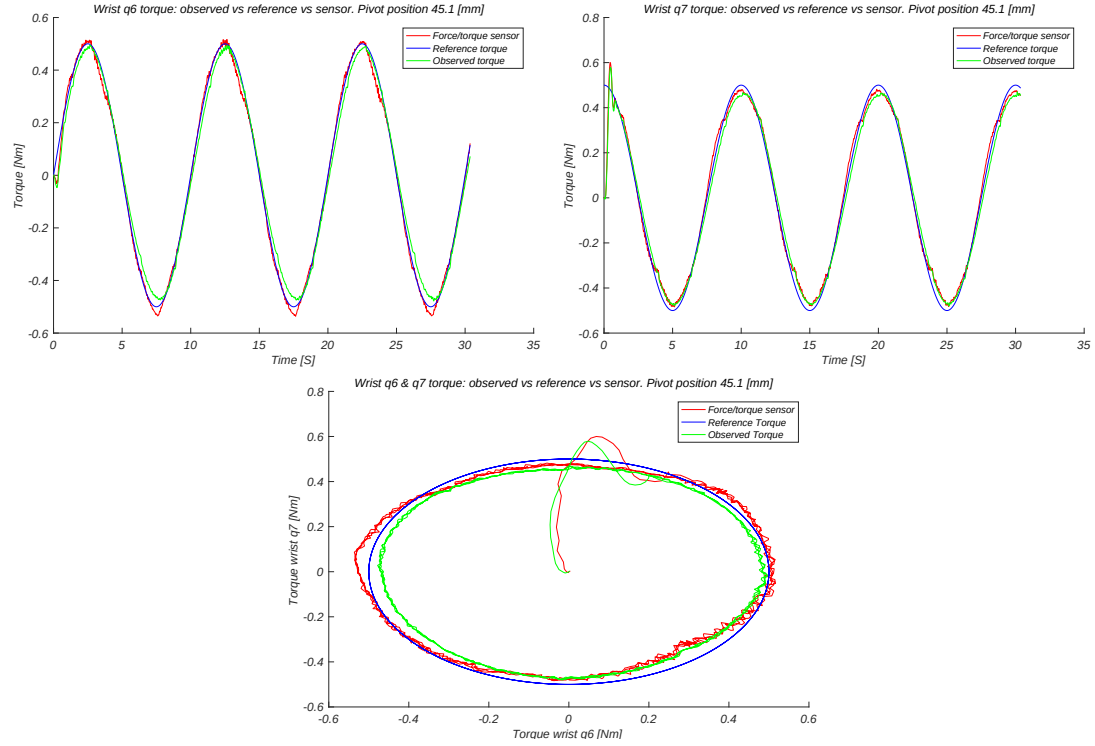


Figure D.8: Validation of the wrist (q6 and q7) with a low stiffness setting.

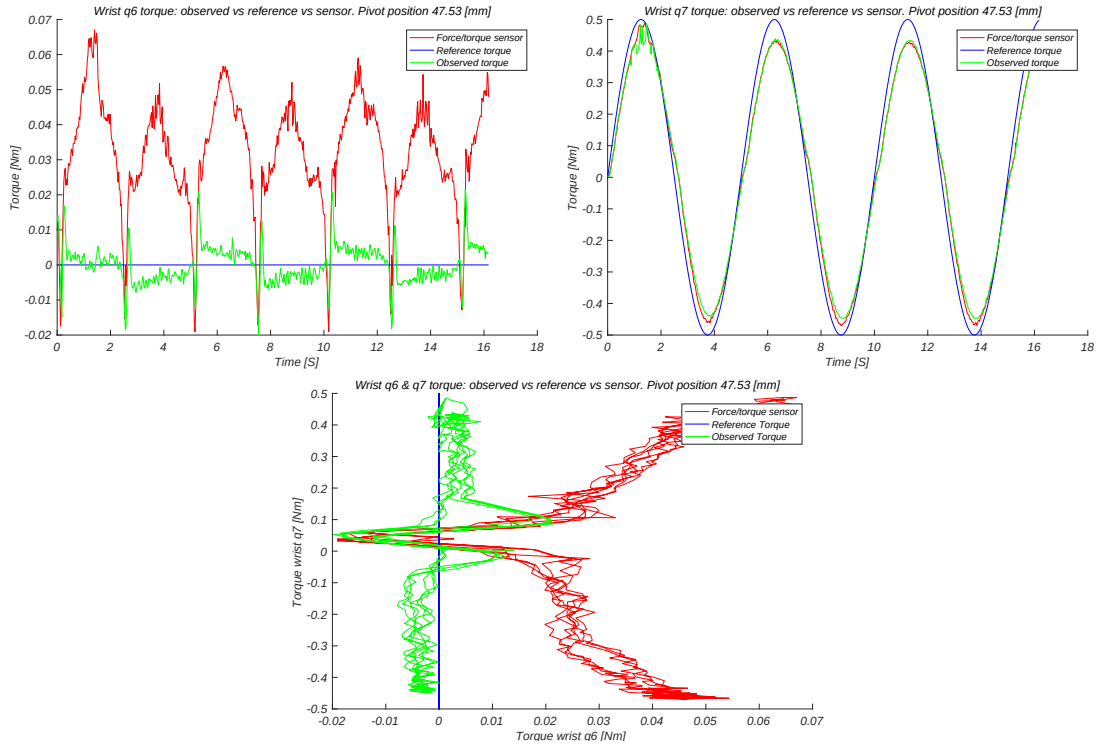


Figure D.9: Validation of the wrist (q6 and q7) with a low stiffness setting.

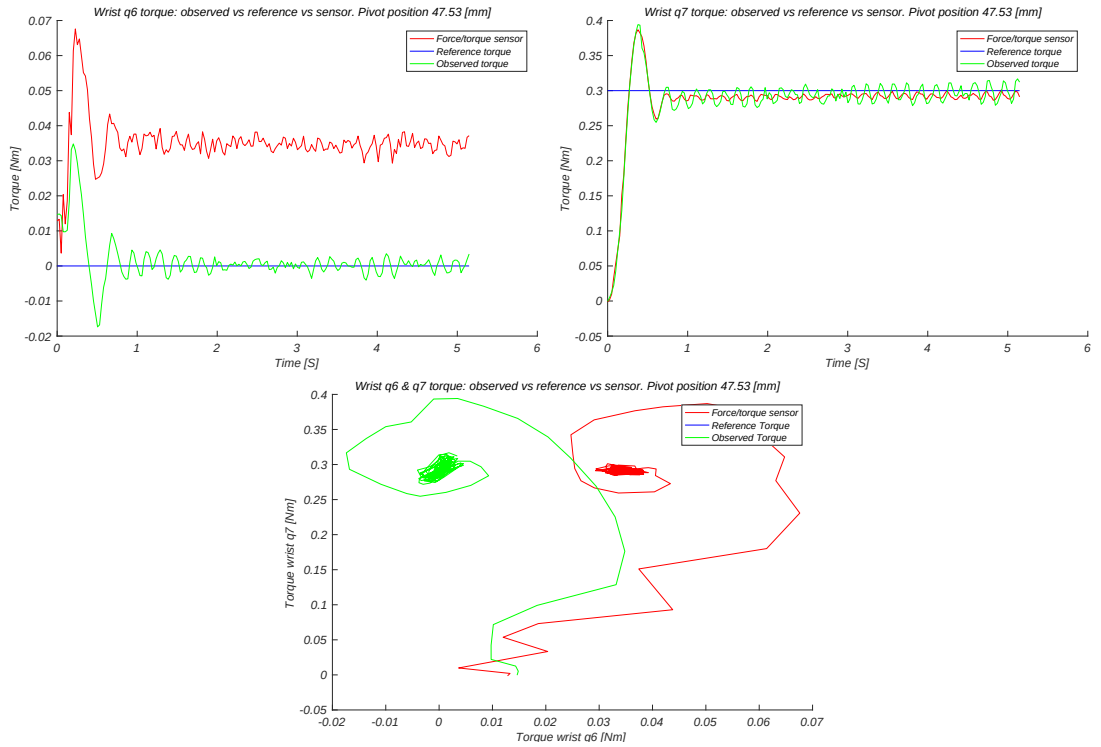


Figure D.10: Validation of the wrist (q6 and q7) with a low stiffness setting.

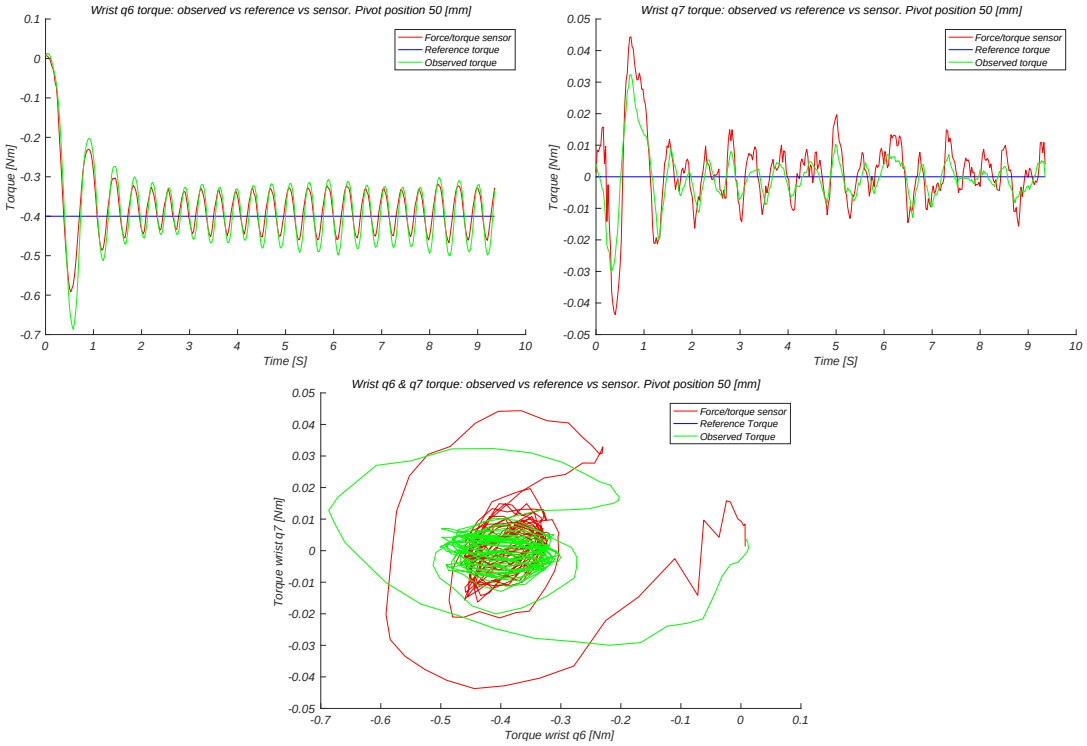


Figure D.11: Validation of the wrist (q6 and q7) with a low stiffness setting.

Bibliography

20sim, s. (2017), 20-Sim Home Page, http://www.20sim.com/.

ATI, A. I. A. (2017), ATI Industrial Automation: F/T Sensor Mini40, http://www.ati-ia.com/products/ft/ft_models.aspx?id=Mini40.

Barbieri, G. B. (2016), Control Architecture for Docking UAVs with a 7-DOF Manipulato, Msc report 045ram2016, University of Twente, Netherlands, https://www.ram.ewi.utwente.nl/aigaion/attachments/single/1353.

Barrett, E., M. Fumagalli and R. Carloni (2016), Elastic Energy Storage in Leaf Springs for a Lever-Arm Based Variable Stiffness Actuator, in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 537–542, doi:10.1109/IROS.2016.7759105.

Barrett, E., M. Fumagalli, R. Carloni, G. Barbieri and M. Reiling (2017), Mechatronic Design of a Variable Stiffness Robotic Arm, in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Boterenbrood, W. (2015), Development of the Low-Level Software Architecture for the Sherpa Robot Arm, Bsc report 024ram2015, University of Twente.

Chai, T. and R. R. Draxler (2014), Root Mean Square Error (RMSE) or Mean Absolute Error (MAE)? – Arguments against Avoiding RMSE in the Literature, **vol. 7**, no.3, pp. 1247–1250, ISSN 1991-9603, doi:10.5194/gmd-7-1247-2014, http://www.geosci-model-dev.net/7/1247/2014/.

Chapra, S. C. (2006), *Applied Numerical Methods with MATLAB for Engineers and Scientists*, McGraw-Hill Science/Engineering/Math, ISBN 978-0-07-313290-7.

ELMO, C. (2017), Tuning with Elmo Composer, http://www.elmomc.com/training/files/3.1-Software-setup-and-tuning-presentation.php.

Elmo, D. (2017), Servo Drives | Elmo Motion Control, http://www.elmomc.com/products/servo-drives.htm.

Groothuis, S. S., S. Stramigioli and R. Carloni (2015), Compliant Manipulators on Graphs, in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6536–6542, doi:10.1109/IROS.2015.7354311, http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7354311.

Harmonic, D. (2017), CSD-32-100-2UH | Harmonic Drive, http://www.harmonicdrive.net/products/gear-units/gear-units/csd-2uh/csd-32-100-2uh.

Hoffman, J. D. (2001), *Numerical Methods for Engineers and Scientists*, Marcel Dekker, New York, 2nd ed., rev. and expanded edition, ISBN 978-0-8247-0443-8.

Keppler, M., D. Lakatos, C. Ott and A. Albu-Schäffer (2016), A Passivity-Based Controller for Motion Tracking and Damping Assignment for Compliantly Actuated Robots, in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 1521–1528, doi:10.1109/CDC.2016.7798482.

Marconi, L., C. Melchiorri, M. Beetz, D. Pangercic, R. Siegwart, S. Leutenegger, R. Carloni, S. Stramigioli, H. Bruyninckx, P. Doherty, A. Kleiner, V. Lippiello, A. Finzi, B. Siciliano, A. Sala and N. Tomatis (2012), The SHERPA Project: Smart Collaboration between Humans and Ground-Aerial Robots for Improving Rescuing Activities in Alpine Environments, in *2012 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 1–4, doi:10.1109/SSRR.2012.6523905, http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6523905.

MoveIt!, M. (2017), MoveIt! Motion Planning Framework, http://moveit.ros.org/.

Petit, F., A. Daasch and A. Albu-Schäffer (2015a), Backstepping Control of Variable Stiffness Robots, **vol. 23**, no.6, pp. 2195–2202, ISSN 1063-6536, doi:10.1109/TCST.2015.2404894,

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7090964.

Petit, F., A. Dietrich and A. Albu-Schäffer (2015b), Generalizing Torque Control Concepts: Using Well-Established Torque Control Methods on Variable Stiffness Robots, **vol. 22**, no.4, pp. 37–51, ISSN 1070-9932, doi:10.1109/MRA.2015.2476576.

Petit, F. P. (2014), *Analysis and Control of Variable Stiffness Robots*, Doctoral thesis, ETH Zurich, doi:10.3929/ethz-a-010404874, https://www.research-collection.ethz.ch/handle/20.500.11850/155026.

Tan, D. J., D. M. Brouwer, M. Fumagalli and R. Carloni (2017), A 2-DOF Joint With Coupled Variable Output Stiffness, **vol. 2**, no.1, pp. 366–372, ISSN 2377-3766, doi:10.1109/LRA.2016.2631730.

Zult, J. (2016), Modeling and Characterization of a 2-DOF Variable Stiffness Joint, Bsc report 043ram2016, University of Twente.