Intrinsically Passive Control for Ultrasound-guided Robotic Scanning of the Breast

S.O.H. (Stijn) de Looijer

MSc Report

Committee:
Prof. dr.ir. M. Wisse
Prof. dr. ir. H. Vallery
Dr. ir. E. Steur
M.K. Welleweerd, MSc

August 2018

013RAM2018
Robotics and Mechatronics
EE-Math-CS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Intrinsically Passive Control for Ultrasound-guided Robotic Scanning of the Breast

Master's thesis for the degree of

Master of Science
in Mechanical Engineering

at the University of Delft,

to be defended publicly on Tuesday August 21, 2018 at 10:00 AM.

by

Stijn de Looijer

Committee:	Prof. dr. ir. M. Wisse	Delft University of Technology
	Prof. dr. ir. H. Vallery	Delft University of Technology
	Dr. ir. E. Steur	Delft University of Technology
	M. Welleweerd, MSc	University of Twente

TU Delft Delft University of Technology

University of Twente.

RAM Robotics and Mechatronics

There are an endless number of things to discover about robotics.
A lot of it is just too fantastic for people to believe.

Daniel H. Wilson

Abstract

Automated breast ultrasound imaging systems provide the ability to combine the high anatomical detail of preoperative imaging with real-time intraoperative imaging. Ultrasound imaging involves interaction between the transducer and the patient, which requires a compliant behaviour of the robot in manipulating the transducer. We present an energy-based control strategy augmented with a visual servoing algorithm to improve the image quality online. An impedance controller is extended with energy- and power-based safety metrics to adjust the dynamic behavior of the robot and with energy tanks to ensure passivity. The quality of the ultrasound images is represented by a confidence map that is computed online, following a model of ultrasound propagation in soft tissue. Experiments were conducted with the 7 degree-of-freedom KUKA LBR Med articulated robot. First, a port-Hamiltonian model of the robot was created to design and test the controller in simulation. Second, the controller was transferred to the robot controller for experimental validation of the proposed control strategy. Simulational and experimental results have validated the controller design in interaction with the environment. Experiments with an ultrasound phantom have demonstrated that the visual servoing algorithm corrects the probe pose in case of (partial) contact loss between the transducer and the skin surface. This significantly increases the global image quality of the 3D image of the breast. Our approach has resulted in a very promising control strategy for autonomous ultrasound scanning that can be integrated in an automated breast biopsy procedure.

Contents

1 Introduction

1.1 Context

Breast cancer is one of the most diagnosed types of cancer worldwide. In the United States, about one out of eight women are affected by invasive breast cancer over the course of their lifetime. Another 266,120 new cases of invasive breast cancer were expected to be diagnosed in 2018 [12].

X-ray mammography (MG) is nowadays the most dominant imaging technique for evaluation of breast abnormalities, but its sensitivity is significantly reduced in the evaluation of dense breasts [22]. At least 50% of cancers are missed by MG in women with dense breasts [56], which is worrying because 71% of breast cancer occurs in dense-breast women [14].

Several studies have shown that supplemental examination by ultrasound imaging and magnetic resonance imaging (MRI) leads to detection of significantly more and smaller cancers than with MG alone [39]. MRI is with a lesion detection rate of 90 to 99% the most sensitive technique for the detection of breast cancer [13]. However, MRI scanning of the breast requires a contrast medium to be injected, is time-consuming, has high cost and has a relatively low specificity of 30 to 72% correct classifications of benign and malignant tumors. According to [78], handheld ultrasonography (HHUS) has a lesion detection rate of 90.6% and 82.5% correct classifications of benign and malignant tumors. HHUS allows real-time imaging, does not use ionizing radiation and has a relatively low cost, but is operator-dependent and hence hard to reproduce.

When suspicious lesion is detected by MG, MRI or ultrasound imaging, it is usually suggested to the patient to have sample cells extracted for further examination. This medical test is called biopsy. Over the past years there has been an increase in patients who require breast biopsy, mainly because of wider access to population screenings for breast cancer. Intraoperative imaging is used in biopsies to guide the needle to the suspicious lesion. The most common technique for this is the freehand technique [54]. This technique requires highly skilled clinicians who have to use one hand to extract sample cells by needle insertion, and the other hand for displaying the needle and tissue by HHUS. HHUS imaging is well-suited for intraoperative imaging, but the images are in general very noisy and hard to reproduce [60]. Preoperative imaging (e.g. MRI) provides more anatomical detail, but it produces static images that differ from reality at the time of surgery because of organ movement and deformation. Automated breast ultrasound (ABUS) imaging systems can be used to combine the advantages of both preoperative and intraoperative imaging.

Vision-based control (also called visual servoing) is a type of control that uses visual feedback to control the motion of a robot. It can be used to control the ultrasound probe in (intraoperative) ultrasound imaging to acquire ultrasound images of a volume of interest that is localized by preoperative imaging. Multidimensional robotic manipulators provide the dexterity and range that is desired for this task. Current robotic manipulators are primary used in industrial manufacturing, where they have to perform a specific task in a well-defined environment. Hence, they often use position controllers, which gives the manipulators a very stiff dynamic

behaviour [59]. Robotic manipulators are insensitive to fatigue and outperform humans in terms of speed, computational power, replication and precision [28], but position control is not suitable for interactive tasks such as ultrasound scanning.

The goals of ABUS imaging systems are to reduce the radiologist's time per case and to produce a standardized, high quality examination for cancer detection [56]. Several studies have already shown that these goals are feasible. A reduction in radiologist time, higher sensitivity and fewer false-positives were reported for ABUS in comparison to MG in [37] and [38]. In [78] was shown that the diagnostic accuracy of HHUS and ABUS are almost identical. However, ABUS performs better in detecting small lesions and in distinguishing between real lesions and inhomogeneous areas. In [80] it was also shown that pain perception was significantly lower in ABUS scanning compared to MG.

1.2 MURAB Project

This thesis is written on behalf of the MURAB project, which is the acronym for MRI and Ultrasound Robotic Assisted Biopsy [3]. The MURAB project is a joint project between, among others, Radboud umc, KUKA, ZGT and the Robotics and Mechatronics (RaM) laboratory of the University of Twente. The goal of this project is to improve the precision and effectiveness in breast biopsy. A biopsy robot will be designed that combines the high precision of MRI images with the flexibility of real-time ultrasound-guided biopsy. The usage of expensive MRI is reduced to a minimum in the workflow of MURAB, because only one MRI image is made. An autonomous ultrasound scan of the total breast area is made with an ultrasound transducer mounted on the end effector of a robotic arm. The ultrasound transducer is equipped with an acoustically transparent pressure sensor to acquire elastogprahic data of the breast tissue. A technique called Tissue Active SLAM (TAS) is used to reconstruct a (deformed) 3D tissue environment from the 2D ultrasound images and elastographic data, which is required to predict tissue deformations during the needle insertion. The radiologist localizes the target and the robotic arm brings the needle guide to the right position. The radiologist inserts the needle to perform the biopsy.

1.3 Research Objective

We believe that the effectiveness and precision of breast biopsies can be improved by combining the advantages of both MRI and HHUS into a robotic system. MRI provides high anatomical detail and hence is good for localization of a target, but only produces static images. On the other hand, HHUS is well-suited for intraoperative imaging, but ultrasound images are in general very noisy and hard to reproduce. Robotic manipulators provide high precision and are insensitive to fatigue, but the control of interactions is a complex task.

The MURAB project aims to combine MRI, HHUS and robotics into an automated system. The procedure can be divided in three phases: MRI imaging, autonomous ultrasound imaging and needle insertion. This thesis project focuses on the control strategy of autonomous ultrasound imaging. Since the robot is used for medical examination on human patients, it has to guarantee the safety of the patient at all times. This means that the robot has to be compliant for unpredictable disturbances from the patient or the environment. The goal of the scanning procedure for this project is to gather high quality ultrasound images of a volume of interest, which can later be extended with localization of a lesion. In order to optimize the image quality online, an ultrasound-based visual servoing algorithm can be added to the control of the robotic manipulator. Considering this, the research objective of this project is as follows:

Design and implementation of a compliant control strategy for autonomous ultrasound scanning that establishes safe human-robot interaction.

The final goal of this thesis is to deliver a proof of concept for an autonomous ultrasound imaging system that provides safe human-robot interaction and incorporates advanced visual servoing to optimize the quality of the captured images. The milestones that have to be accomplished in order to meet the research objective and to validate the proof of concept are as follows:

- Design and implementation of a compliant control strategy
- Design and implementation of an ultrasound-based visual servoing algorithm

The design of the controller will be examined both in simulation as in experiments on the physical robot. The visual servoing algorithm will first be tested offline and later with the probe attached to the end effector of the manipulator. Since the project is in the experimental phase, we will use a phantom instead of ultrasound scanning on human patients. We will produce our own ultrasound phantom, which gives us the flexibility to create a desired shape and structure for the phantom.

1.4 Contributions of this Thesis

The contributions of this thesis can be described as follows:

- Development of a simulation environment for testing of the controller.

  We use the framework of port-Hamiltonian system theory and screw theory to build a model of the robotic manipulator with the 20-sim software. This allows us to test the controller first in simulation, before implementation on the physical robot.

- Design of a compliant control strategy augmented with an advanced ultrasound-based visual servoing for autonomous ultrasound scanning.

  Previous studies have implemented hybrid control for autonomous ultrasound imaging, teleoperated compliant control for ultrasound imaging, compliant control strategies for other purposes and advanced ultrasound-based visual servoing algorithms for probe manipulation. However, to the author's knowledge, there exists no control strategy that is compliant, allows autonomous scanning and also incorporates ultrasound-based visual servoing to improve the image quality online. We take an energy-based approach to control the robot, which allows us to implement energy-based safety limitations and guarantee the passivity of the overall system.

- Implementation of the control strategy on a physical system.

  Other studies have been addressed in literature that use a similar approach to design a compliant control strategy or to manipulate the probe with ultrasound-based visual servoing. However, there are no successfull attempts that have combined both into one robotic system and also implemented it on a physical system for autonomous ultrasound scanning. We will implement our control strategy on the medically certified KUKA LBR Med, which is a 7-DOF redundant robotic manipulator. The control strategy is designed for autonomous scanning of the breast, but can also be applied for scanning of other body parts.

1.5 Related Work

In this section we will discuss related work on autonomous ultrasound imaging, compliant control of robotic manipulators and ultrasound-based visual servoing that has been addressed in the literature.

Autonomous ultrasound imaging can be controlled by minimizing e.g. a visual error, position error, force error or a combination of several errors. In [61] a 7-DOF serial manipulator, with force feedback from a 6-DOF force-torque sensor, was used to control the contact force and

the orientation of the probe with respect to the skin. Position and force were controlled by a PID-controller. In [23], a 7-DOF teleoperated serial manipulator with force controller is used to maintain a constant contact force between the patient's skin and the probe. In [34], a hybrid position/force control algorithm is combined with a pixel intensity based ultrasound image analysis to control a 6-DOF serial manipulator. A PI-controller was used for both position control and force control. In [51], a hybrid force/vision control strategy was implemented for ultrasound scanning of an abdominal phantom. The intensity of the pixels in the image was taken as visual input to control a 6-DOF robotic manipulator. These systems all use a form of PID-control, where high gains can be injected to achieve minimal error. This type of control is not guaranteed to be stable and is therefore tricky to implement for interactions with human patients. Also, the pixel intensity based image analysis that is used in [34] and [51] does not provide a lot of information about the image quality.

According to [24], an interactive control strategy - that gives the manipulator a certain compliance - should be implemented for autonomous ultrasound imaging. In [57], a 7-DOF robot manipulator was controlled in task space, while guaranteeing a compliant behavior for the redundant DOFs by implementing null-space compliance. Performance on the task was maintained under safe collisions between the robot's body and the environment, as well as under disturbances on the end effector. A controller design for human-friendly robotic manipulators that incorporates safety and performance issues was proposed in [63]. A safety aware intrinsically passivity controller was implemented on top of the impedance controller, that modifies the dynamic behaviour of the manipulator based on the amount of power consumption and the total energy of the manipulator. Passivity is guaranteed by virtually limiting the available energy of the joint actuators. Experiments were performed on a 1-DOF setup. In [18], this was extended to the control of a 7-DOF manipulator. Simulation experiments have shown the applicability of this control for the control of interaction with a multidimensional manipulator.

Ultrasound-based visual servoing can be used to manipulate the probe by using visual features that are extracted from the ultrasound images. A distinction can be made between algorithms that use global image characteristics (e.g. [34], [51], [19]) and algorithms that aim for localization of a target (e.g. [51], [50], [47]). Localization of a target often requires additional post-processing like image segmentation or denoising. A more detailed discussion of the state-of-the-art ultrasound-based visual servoing algorithms will be provided in chapter 6.2. For now it is important to note that the more advanced visual servoing algorithms are described in terms of probe control, but no coupling has yet been made to manipulator control.

Two thesis projects ([61] and [74]) within MURAB have been previously devoted to autonomous ultrasound imaging with the KUKA LWR4+ manipulator. In their approach they implemented a hybrid position/force control, where position control was used to follow a predefined trajectory and force control was used to maintain a constant contact force between the probe and the patient's skin. We will take another approach by implementing an energy-based control strategy.

1.6 Report Outline

This report is organized as follows:

- **Chapter 2** describes the basic concepts of ultrasound imaging and their relation with ultrasound image quality.
- **Chapter 3** describes the terminology and main mathematical concepts of robotic manipulators.
- **Chapter 4** describes the control of robotic manipulators in interaction with the environment.

- **Chapter 5** describes the concept of passivity and how it can be implemented in the control of robotic manipulators.
- **Chapter 6** describes the concept of ultrasound-based visual servoing and the implementation for autonomous ultrasound imaging.
- **Chapter 7** describes the derivation of a port-Hamiltonian model of a robotic manipulator that can be used in the simulation experiments.
- **Chapter 8** describes the system specifications, including both hardware and software, and the design and production of the ultrasound phantom.
- **Chapter 9** describes the design of the experiments.
- **Chapter 10** describes the simulation model of the robotic manipulator and it shows simulation results to examine the performance of the controller.
- **Chapter 11** discusses the results of the experiments.
- **Chapter 12** concludes this work and provides recommendations for ongoing work.

2 Ultrasound Imaging

This chapter provides an introduction to the basics of ultrasound imaging. This theory will be used to include ultrasound image quality assessment in the design of an ultrasound-based control strategy. The concept of ultrasound imaging is explained first in order to understand how different control actions can influence the ultrasound image quality.

2.1 Basic Principles of Ultrasound Imaging

This section provides an introduction to the basic principles of ultrasound imaging. The quality of raw ultrasound images depends on the design and settings of the ultrasound device, as well as on extrinsic factors such as contact force, probe orientation and the distribution of ultrasound gel [19]. This study will only focus on the control of extrinsic factors. The design of the ultrasound device will not be examined and the settings of the ultrasound device are assumed to be fixed.

Ultrasound imaging is a common technique for diagnostic and intraoperative imaging. Ultrasound waves are generated by a piezoelectric transducer array that is placed on the skin. The transmission of ultrasound waves between air and the skin is very low, because these media have a large difference in acoustic impedance. To enhance transmission of the ultrasound waves into the tissue, a water-based gel - with an acoustic impedance that is very close to that of the skin - is usually applied to the skin surface. The principle of ultrasound imaging is illustrated in figure 2.1. The piezoelectic elements in the ultrasound transducer produce ultrasound waves and listen to the echoes reflected by regions of different echogenicity, i.e. the ability of a tissue to reflect or transmit ultrasound waves [33]. The reflected ultrasound waves are converted back to electric signals, which are processed to form a 2D ultrasound image.

Anatomical areas can be characterized as anechoic, hypoechoic and hyperechoic. They appear on the image as black, grey and white respectively. White areas represent echoes that are

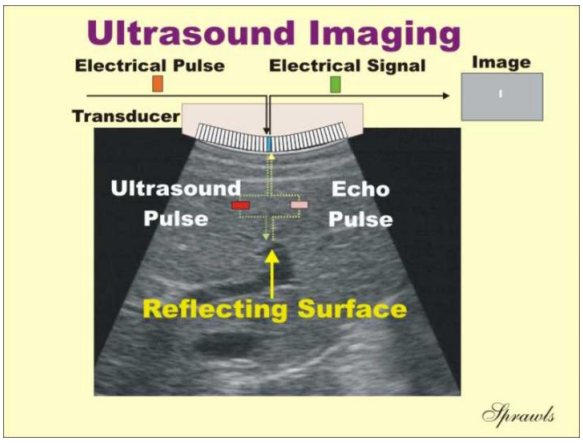


Figure 2.1: Principle of ultrasound imaging [11].

produced by surfaces or boundaries between different types of tissue. When there is no reflection inside an area, it appears as black in the ultrasound image. This happens in fluids (e.g. a cyst), because there are no reflecting surfaces inside the fluid. Most anatomical areas are composed of a mixture of different tissue that appear as different shades of gray in the ultrasound image, depending on the amount of reflection inside the tissue. Another important tissue property is anisotropy. In an anisotropic tissue, the reflection of ultrasound waves can change tremendously with a small mutation in the angle of incidence. When the probe is normal to the skin surface, in general more ultrasound waves will be reflected back to the transducer and fewer will be scattered away, which results in a better image.

2.2 Ultrasound Image Quality

One of the main contributions of this project is to include online ultrasound image quality assessment in the control of the robot. This section provides an overview of measures to describe ultrasound image quality and their relation with the control actions that can be carried out by the robot.

Ultrasound images are in general very noisy. The most prominent noise sources in the diagnosis of ultrasound images are speckle noise, blurring, attenuation and system noise [27] (figure 2.2). Speckle noise and blurring happen at low ultrasound frequency, while system noise and attenuation happen at high frequencies. Attenuation is caused by scattering and absorption of the acoustic signal. An attenuated signal can be amplified, but this is accompanied by amplification of signal noise. Speckle noise, which appears as granular structures in different size and intensity, is a typical time invariant noise source of ultrasound images. It is caused by particles that are smaller than the ultrasound modality, which produces a random interference pattern. Speckle noise can be reduced by denoising algorithms. However, a problem of most denoising algorithms for ultrasound images is that they blur the image, which reduces the resolution of the images.

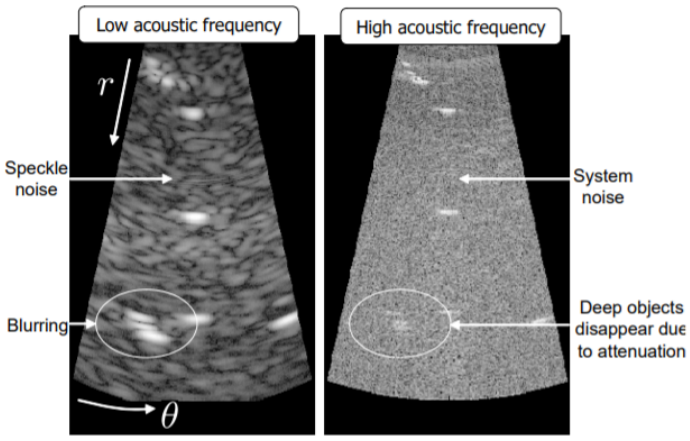


Figure 2.2: Noise sources in ultrasound images /citeErez2006.

It is clear that the described noise sources deteriorate the quality of ultrasound images, but they do not provide a quantifiable measure of image quality. According to the literature (e.g. [76] [6] [52]), the most common quantifiable measures to describe image quality are contrast resolution, sensitivity, temporal resolution and detail resolution. Contrast resolution is known as the ability to distinguish soft tissue regions by a difference in echogenicity and is very dependent on the amount of speckle noise. Sensitivity is a measure of the minimum detectable echogenicity. Temporal resolution is a measure of the fastest speed at which the ultrasound system can follow moving objects and detail resolution is known as the minimum distance at which two points in an ultrasound image can be distinguished. For the interested reader, an

exhaustive description of the aforementioned measures including formulas can be found in [76].

After acquisition of the ultrasound images, their quality can be further enhanced by post-processing techniques such as despeckling filters [44] and spatial compounding [75]. The focus of this study is on acquiring raw high-quality ultrasound images, so image enhancement by post-processing will not be considered here.

Summarizing, the quality of an ultrasound image is reduced by speckle noise, attenuation, system noise and blurring and can be defined by contrast resolution, sensitivity, temporal resolution and detail resolution. In chapter 6 we will perform a short literature review on ultrasound-based visual servoing algorithms to find out which measure is most appropriate for an ultrasound-based control.

3 | Mathematics of Robotic Manipulators

This chapter explains the terminology and main mathematical concepts of robotic manipulators, which can be used to design a control strategy that results in a desired behaviour of the manipulator for autonomous ultrasound scanning. Chapter 3.1 explains the kinematics of serial manipulators, which describes motions of the manipulator without taking forces into account. Section 3.2 explains the dynamics of serial manipulators, which also describes interactions and the consequences of external forces. We will use the framework of screw theory, which was introduced by [15] and employed by [69] in the field of robotic manipulators. Furthermore, we will use the concepts of Lie Groups [31]. Definitions of the fundamental concepts of Lie Groups are given in appendix A. For the interested reader, we would like to refer to [67] and [65] as a reference for this section.

3.1 Kinematics of Serial Manipulators

This section describes the kinematics of serial manipulators, which can be categorized into direct kinematics and differential kinematics. Direct kinematics is used to compute the pose of the end effector from the joint positions, while differential kinematics are used to relate joint velocities to motions of the end effector. The term forward kinematics is used for calculations from joint space to Cartesian space, while inverse kinematics works the other way around.

3.1.1 Direct Kinematics

An n-link serial manipulator consists of a base, n joints and n links. Let us define frames $\Psi_0 ... \Psi_{ee}$ as illustrated in figure 3.1, with Ψ_0 the static base frame and Ψ_{ee} the end effector frame.

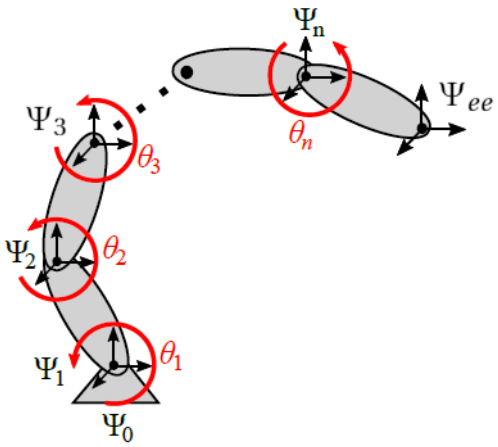


Figure 3.1: Serial kinematic chain of a robotic manipulator [18].

A point expressed in coordinate frame Ψ_i with respect to the frame Ψ_j can be expressed as:

$$\underbrace{\begin{pmatrix} p^j \\ 1 \end{pmatrix}}_{P_j} = \underbrace{\begin{pmatrix} R_i^j & p_i^j \\ 0_{1x3} & 1 \end{pmatrix}}_{H_i^j} \underbrace{\begin{pmatrix} p^i \\ 1 \end{pmatrix}}_{P_i} \tag{3.1}$$

Where $p^i \in \mathbb{R}^3$ is the point expressed in frame Ψ_i, p^j is the point expressed in frame Ψ_j, $P_i \in \mathbb{R}^4$ is the projective space of p^i, P_j is the projective space of p^j, $p_i^j \in \mathbb{R}^3$ the vector from the origin of frame Ψ_j to the origin of frame Ψ_i and $R_i^j \in \mathbb{R}^{3\times3}$ is the rotation matrix from Ψ_i to Ψ_j. R_i^j belongs to the Special Orthonormal Group SO(3), which is a Lie group of orthonormal matrices with determinant 1:

$$SO(3) = \left\{ \mathbf{R} \in \mathbb{R}^{3\times3}; R^{-1} = R^T; det(R) = 1 \right\} \tag{3.2}$$

The matrix $H_i^j \in \mathbb{R}^{4\times4}$ is the homogeneous transformation matrix that defines the pose of frame Ψ_i with respect to frame Ψ_j and belongs to the Special Eucledian Group SE(3), which is also a Lie group:

$$SE(3) = \left\{ \begin{pmatrix} R_i^j & p_i^j \\ 0_{1\times3} & 1 \end{pmatrix} : R_i^j \in SO(3), p_i^j \in \mathbb{R}^3 \right\} \tag{3.3}$$

The pose, i.e. the position and orientation, of frame Ψ_n with respect to the base frame Ψ_0 is given by the serial kinematic chain:

$$H_n^0 = H_1^0 H_2^1 ... H_n^{n-1} \tag{3.4}$$

According to Mozzi's theorem [49], any motion of a rigid body can be described by a rotation around a screw axis and a translation along the same axis. In screw theory, this motion can be represented by a twist, which can be expressed in vector form or matrix form as:

$$T = \begin{pmatrix} \omega \\ v \end{pmatrix} = \begin{pmatrix} w \\ r \wedge w \end{pmatrix} + \lambda \begin{pmatrix} 0 \\ \omega \end{pmatrix} \qquad \tilde{T} = \begin{pmatrix} \tilde{\omega} & v \\ 0 & 0 \end{pmatrix} \tag{3.5}$$

Where $\omega \in \mathbb{R}^3$ defines the rotation around the screw axis and $v \in \mathbb{R}^3$ defines the translation along the screw axis. The vector r and scalar λ are defined as illustrated in figure 3.3. The tilde form of ω is given by a skew-symmetric matrix:

$$\tilde{\omega} = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \tag{3.6}$$

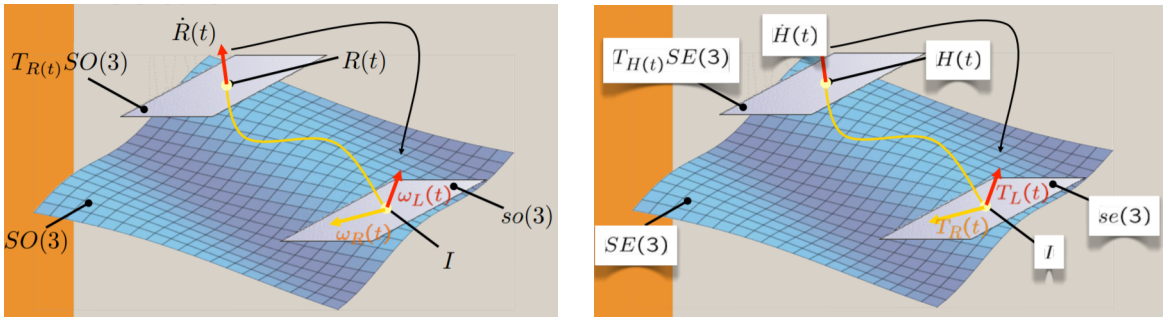
Matrices $\tilde{\omega}$ belong to so(3), which is the Lie algebra of SO(3):

$$so(3) = \{ \tilde{\omega} \in \mathbb{R}^{3\times3} : -\tilde{\omega} = \omega^T \} \tag{3.7}$$

Matrices $\tilde{T}$ belong to se(3), which is the Lie algebra of SE(3):

$$se(3) = \left\{ \begin{pmatrix} \tilde{\omega} & v \\ 0 & 0 \end{pmatrix} : \tilde{\omega} \in so(3), v \in \mathbb{R}^3 \right\} \tag{3.8}$$

A graphical representation of SO(3), so(3), SE(3) and se(3) is given in figure 3.2.



(a) **(b)**

Figure 3.2: Graphical representation of **(a)** the Special Orthonormal group SO(3) and **(b)** the Special Eucledian group SE(3) [69].

The relation between twists in different frames is specified by an adjoint:

$$Ad_{H_i^j} = \begin{pmatrix} R_i^j & 0 \\ \tilde{p}_i^j R_i^j & R_i^j \end{pmatrix} \tag{3.9}$$

Where $\tilde{p}_i^j$ is the tilde form of the translation vector p_i^j. The coordinate transformation of the twist of frame Ψ_i with respect to the base frame Ψ_0, expressed in frame Ψ_i, to frame Ψ_j is given by:

$$T_i^{j,0} = Ad_{H_i^j} T_i^{i,0} \tag{3.10}$$

Where $T_a^{c,b}$ corresponds to the twist of frame a with respect to frame b, expressed in frame c.

Equation (3.4) can be rewritten in the following form:

$$H_n^0(q_1, q2, ..., q_n) = e^{\tilde{\hat{T}}_1^{0,0} q_1} e^{\tilde{\hat{T}}_2^{0,1} q_2} ... e^{\tilde{\hat{T}}_n^{0,(n-1)} q_n} H_n^0(0) \tag{3.11}$$

With $\hat{T}_i^{0,(i-1)}$ constant unit vectors of twists in the initial configuration $q_1, q_2, ..., q_n = 0$, which are given by:

$$\hat{T}_i^{0,(i-1)} = \begin{pmatrix} 0 \\ r \wedge \hat{\omega}_i^{0,(i-1)} + \lambda \omega_i^{0,(i-1)} \end{pmatrix} \tag{3.12}$$

In case of a pure translation, and otherwise as:

$$\hat{T}_i^{0,(i-1)} = \begin{pmatrix} \hat{\omega}_i^{0,(i-1)} \\ r \wedge \hat{\omega}_i^{0,(i-1)} + \lambda \omega_i^{0,(i-1)} \end{pmatrix} \tag{3.13}$$

With $\hat{\omega}_i^{0,(i-1)}$ a unit vector of angular velocity. The scalar r is the distance from the origin of frame Ψ_i to the screw axis and λ is the translation on the screw axis due to one revolution. This is illustrated in figure 3.3.

Equation 3.11 is known as Brockett's formula and it represents the direct kinematics of the serial manipulator. It computes the pose of the end effector from the joint positions only (and the fixed dimensions of the links), so it is basically a mapping from joint space to SE(3):

$$L : Q \rightarrow SE(3); \quad q \mapsto H_n^0(q) \tag{3.14}$$

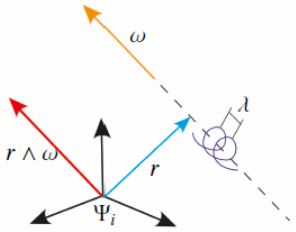


Figure 3.3: Representation of the parameters that define a twist [42].

3.1.2 Differential Kinematics

Differential kinematics relate the joint velocities to motions of the end effector:

$$T_n^{0,0} = J(q)\dot{q} \tag{3.15}$$

Where $J(q) \in \mathbb{R}^{6\times n}$ is the manipulator jacobian given by:

$$J(q) = (T_1 \; T_2 ... T_n) \tag{3.16}$$

With:

$$T_i = Ad_{H_{i-1}^0} \hat{T}_i^{(i-1),(i-1)} \tag{3.17}$$

Calculation of the joint velocities from equation (3.15), requires the inverse of the jacobian: $\dot{q} = J^{-1}(q)T_n^{0,0}$. The kinematic map of a redundant manipulator is a surjective, not injective function, which means that the solution of $\dot{q}$ is not unique. Furthermore, it can happen that the rank of $J(q)$ decreases for a certain configuration q. In that case, $J^{-1}(q)$ can become very big which leads to high joint velocities and possibly unstable behavior, which must be avoided.

3.2 Dynamics of Serial Manipulators

This section describes the dynamics of serial manipulators, which includes the effect of forces and torques on the motion of the manipulator. According to Poinsot's theorem, any system of forces can be expressed as a pure linear force along a line and a pure moment around that same line. In screw theory this can be represented by a wrench, which is the dual of a twist. A wrench is given in vector form as a 6-dimensional row vector:

$$W = (\tau,\ F) = \underbrace{(r \wedge F,\ F)}_{force} + \underbrace{\lambda(F,\ 0)}_{moment} \tag{3.18}$$

Where $F \in \mathbb{R}^3$ defines a pure linear force along the screw axis, $\tau \in \mathbb{R}^3$ defines a torque around the screw axis and λ and r according to figure 3.3. A wrench is a co-vector, which is a linear operator that maps vectors to real numbers:

$$P = WT \tag{3.19}$$

Where P is the power, which is a scalar that is independent of the coordinate choice. T is again the twist as defined in equation (3.5). The coordinate transformation of a wrench expressed in frame Ψ_j to the wrench expressed in frame Ψ_i is given by:

$$(W^i)^T = Ad_{H_i^j}^T (W^j)^T \tag{3.20}$$

The Lie group structure allows to describe motions independently of the configuration of the body. This is essential to talk about the interconnection through power ports of bodies with different poses, which will later be used to describe the dynamic model of the manipulator.

3.2.1 Equations of Motion of a Rigid Body

In this section we will apply the Newton-Euler equations to derive the equations of motion of a rigid body. These equations relate the motion of the principal inertia frame of a rigid body with the sum of forces and torques acting on the rigid body. The resulting equation is consistent with the principles of screw theory.

It can be shown that any rigid body behaves as a uniform ellipsoid of mass m and with principal moments of inertia j_x, j_y, j_z, centered and oriented at the principal inertia frame Ψ_k [69]. The (0,2) inertia tensor of a rigid body, expressed in frame Ψ_k, is given by:

$$\mathscr{I}^k = \begin{pmatrix} J & 0 \\ 0 & mI \end{pmatrix} \tag{3.21}$$

Where $J \in \mathbb{R}^{3\times3}$ is a diagonal matrix with the principal moments of inertia on its diagonal and I is the identity matrix. Similar to Euler's first law of motion ($p = mv$), the dynamics of a rigid body can be expressed by means of a momentum screw $(\mathscr{P})^T$ [71] [69]:

$$(\mathscr{P})^T = \mathscr{I}T \tag{3.22}$$

Similar to the rate of change of momentum ($\dot{p} = F$), the rate of change of momentum screw $\dot{\mathscr{P}}^0$ expressed in frame Ψ_0 is given by:

$$\dot{\mathscr{P}}^0 = W^0 \tag{3.23}$$

The following expression can be used to describe the motion in the principal inertia frame Ψ_k:

$$(\dot{\mathscr{P}}^k)^T = \underbrace{ad^T_{T^{k,0}_k}(\mathscr{P}^k)^T}_{Coriolis forces} + (W^k)^T \tag{3.24}$$

With:

$$ad^T_{T^{k,0}_k} = \begin{pmatrix} \tilde{\omega}^{k,0}_k & 0 \\ \tilde{v}^{k,0}_k & \tilde{\omega}^{k,0}_k \end{pmatrix} \tag{3.25}$$

Equation (3.24) can be rewritten as:

$$(\dot{\mathscr{P}}^k)^T = \mathscr{I}^k \dot{T}^{k,0}_k + \underbrace{\dot{\mathscr{I}} T^{k,0}_k}_{=0} = \begin{pmatrix} -\tilde{\omega}^{k,0}_k & -\tilde{v}^{k,0}_k \\ 0 & -\tilde{\omega}^{k,0}_k \end{pmatrix} \mathscr{I}^k T^{k,0}_k + (W^k)^T \tag{3.26}$$

Equation (3.26) can be rewritten to describe the motion of a rigid body due to an external wrench [67]:

$$\begin{pmatrix} J\dot{\omega}^{k,0}_k \\ mI\dot{v}^{k,0}_k \end{pmatrix} + \begin{pmatrix} \omega^{k,0}_k \times J\omega^{k,0}_k \\ \omega^{k,0}_k \times mIv^{k,0}_k \end{pmatrix} = \begin{pmatrix} (\tau^k)^T \\ (f^k)^T \end{pmatrix} \tag{3.27}$$

This expression will be useful in modeling the interaction between the manipulator and its environment and will also be used in chapter 4.2.4 to control the robot.

3.2.2 General Dynamics of Robotic Manipulators

A serial robotic manipulator can be considered as a system of rigid bodies which are interconnected by joints. The joints introduce constraints for the relative motion of the rigid bodies. The equations of motion of the manipulator can e.g. be derived by combining Newton's second law and Euler's second law with these constraints. The equations of motion of an n-link rigid-body manipulator can, in their most simplistic canonical form, be written as the following non-linear second-order differential equation:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q)^T = \tau^T \tag{3.28}$$

where q, $\dot{q}$ and $\ddot{q} \in \mathbb{R}^n$ are vectors of respectively generalized joint position, velocity and acceleration, $M(q) \in \mathbb{R}^{n\times n}$ is a symmetric and positive definite mass matrix, $C(q,\dot{q})\dot{q} \in \mathbb{R}^n$ is a co-vector of Coriolis and centrifugal torques, $G(q) \in \mathbb{R}^n$ is a co-vector of gravitational torques and $\tau \in \mathbb{R}^n$ is a co-vector of equivalent joint torques due to motor torques and interaction forces [57] [57] [58].

In this chapter we described the kinematics and dynamics of serial manipulators by the concepts of screw theory and Lie groups. We derived equations to describe the motion of the manipulator as a result of joint torques and external wrenches. In the next chapters, we will use these equations to design a control strategy that results in a desired behaviour of the manipulator for autonomous ultrasound scanning.

4 Interaction Control of Robotic Manipulators

This chapter explains the design and importance of an interactive control strategy for autonomous ultrasound scanning. A control strategy that is well-suited for interactive tasks is impedance control. This chapter describes the implementation of an energy-based impedance controller, starting with a one-dimensional system and extending this to a multidimensional serial manipulator. At the end of the chapter, energy and power limitations are included in the design of the impedance controller in order to establish safe human-robot interaction.

4.1 Dynamical Interaction

In many industrial applications, robots are isolated from unpredictable circumstances in their environment to perform a specific task. However, current developments in robotics more often require robots to interact with the environment in situations that can not be predicted. This section describes the importance of interactive control strategies in tasks that include dynamical interaction. According to [67], interaction is defined as:

"Given two systems A and B, we say that the systems interact if the behavior of A is influenced by the presence of B and/or vice versa."

In autonomous ultrasound imaging, the manipulator is in dynamical interaction with the patient, due to breathing and other unavoidable movements of the patient and deformation of the scanned tissue. The interaction between those subsystems is characterized by the exchange of energy between them. According to [32], a manipulator may no longer be treated as an isolated system when it is in dynamical interaction with the environment. Control of a vector quantity - as in force control or position control - is inadequate, because these strategies are not capable of controlling the mechanical work exchanged between the probe and the patient. For example, when the probe deviates from its desired trajectory in position control, because of breathing motions of the patient, high gains will be injected to follow the desired trajectory. This can cause high forces and torques on the patient's skin, which is unacceptable in medical applications. Instead, the interaction with the environment can be controlled by adjusting the dynamics of the robot. This allows to control the robot in a power-consistent way, where energy is exchanged between the robot and the environment through so-called bidirectional power ports. This concept will be discussed in more detail in chapter 5.

4.2 Impedance Control

A control strategy that is very well-suited for the control of interaction is impedance control. Impedance control is a type of physical control, in which the controlled robot acts like a physical system. Instead of controlling just one state variable (e.g. force, position, velocity), a dynamic relationship between those variables is established by controlling the impedance of the manipulator [32]. By controlling the impedance of the controller, desirable behavior between the manipulator and the environment can be achieved. We believe that this is a

promising approach for autonomous ultrasound imaging, in which the ultrasound probe has to interact with the patient in a compliant way.

4.2.1 Simple Impedance Controller

We refer to the *simple impedance controller* as the 1D controller illustrated in figure 4.1. This system corresponds to a second order system mass-spring-damper system with mass m, spring stiffness k_c and damping coefficient b. The controller aims for minimizing the error between the current position x of mass m and the desired position x_d. It is equivalent to a traditional Proportional-Derivative (PD) controller, where the stiffness k_c corresponds to the proportional gain K_p and the damping coefficient b corresponds to the derivative gain K_D.

The dynamic equation of the mass-spring-damper system of figure 4.1 is given by:

$$m\ddot{x} + k_c(x_d - x) + b\dot{x} = F_{ext} \tag{4.1}$$

With F_{ext} the external forces acting on mass m. Control of this system requires measurements of the position x and velocity $\dot{x}$ of mass m. However, velocity measurements are often not available. The velocity can be calculated from position measurements by numerical differentiation, by using state variable filters or by using observers. However, all these methods are not very suited for physical control: the differentiation increases errors due to noise in the position measurements, state variable filters do not have any direct physical interpretation and observers require a model of the system and the environment [66]. We propose to use the damping injection framework [53] to circumvent the need of velocity measurements and to create damping in order to ensure a strictly passive behavior of the overall system. The concept of passivity and its importance to control of interaction will be discussed in chapter 5.

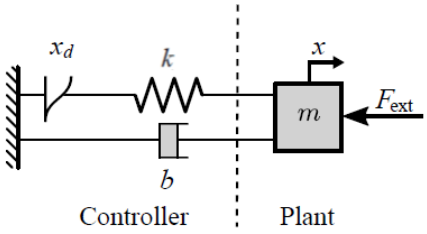


Figure 4.1: Simple impedance controller with stiffness k and damping coefficient b [18].

4.2.2 Energy Shaping

The simple impedance controller implements a virtual spring and damper to move the end effector to a desired position x_d. This introduces a potential energy function $V(r)$ and Rayleigh function $\mathscr{R}(r)$, where $r = x - x_d$, such that:

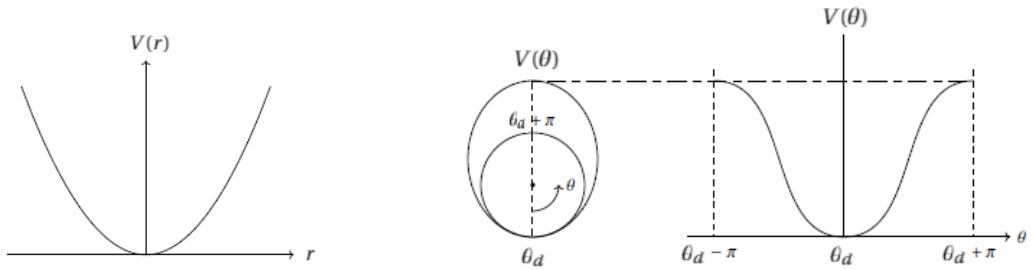
$$\begin{aligned} V(r) &= \frac{1}{2}kr^2 \\ \mathscr{R}(r) &= \frac{1}{2}b\dot{r}^2 \end{aligned} \tag{4.2}$$

The controller force F_c acting on mass m, for the case that x_d is constant and $F_{ext} = 0$, is given by:

$$\begin{aligned} F &= -\frac{\partial V(r)}{\partial r} - \frac{\partial \mathscr{R}}{\partial \dot{r}} \\ &= -k(x - x_d) - b\dot{x} \end{aligned} \tag{4.3}$$

This method is called energy shaping, because the potential energy is shaped by implementing the spring and damper. The potential energy function $V(r)$ is a positive semi-definite function with a minimum in $r = 0$, such that it drives the mass m to the desired position x_d. In general, any positive semi-definite potential energy function can be implemented as long as the system is backdrivable, provides position measurements and has low friction [69].

In this study, we will use energy functions that correspond to translational and rotational springs for respectively translations and rotations of the end effector. The potential energy function of a translational spring is given in figure 4.2a. The potential energy function of a rotational spring $V(\theta)$ is a bit more complex, because a rotation of the end effector with $\theta = 2\pi$ results in the same configuration. Since the controller has to apply the same torque for the equivalent configurations, the potential energy function of a rotational spring has to be closed as a circle. This is illustrated in figure 4.2b, where θ_d is the desired rotation.



(a) **(b)**

Figure 4.2: Potential energy function of **(a)** a translational spring and **(b)** a rotational spring [42].

4.2.3 Damping Injection Framework

The concept of damping injection for the one-dimensional case is illustrated in figure 4.3. The controller is extended with a virtual controller mass m_c and controller spring k_c. Velocity measures of m_c are available from the state of the controller, which allows the damper to be implemented by only measuring joint positions of the manipulator. If m_c and k_c are chosen such that $m_c << m$ and $k_c >> k$, the system behaves again like a second order system with mass m, damping b and stiffness k_c [66].

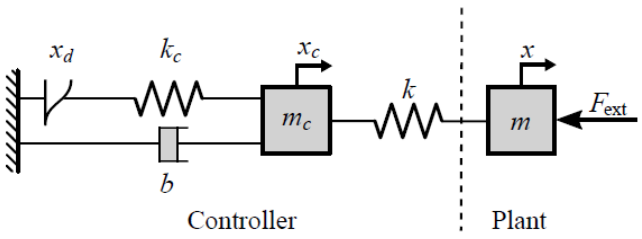


Figure 4.3: Concept of damping injection [66].

The damping injection framework also allows to passively handle saturation of actuator torques. If it is known that an actuator has a certain saturation value F_{sat}, it is possible to implement a passive non-linear controller spring k_c as illustrated in figure 4.4. This ensures that both the controller mass m_c and robot mass m will never receive a force from the actuator that is higher than F_{sat} [66].

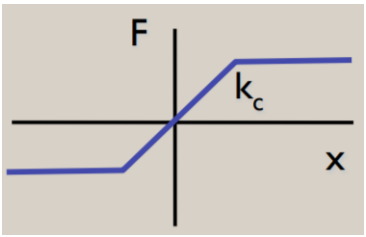


Figure 4.4: Passive non-linear spring to account for saturation of actuators [69].

4.2.4 Extension to Multidimensional Case

So far we discussed the implementation of an impedance controller for a one-dimensional system. This concept can be extended to the control of a multi-DOF serial manipulator. A multidimensional virtual spring $K \in \mathbb{R}^{6\times6}$ can be connected between the end effector, defined by its pose H^0_{ee}, and the desired pose H^0_d. Damping can be added parallel to the spatial spring by a multidimensional damper $B \in \mathbb{R}^{6\times6}$ or in joint space by a damper b_n on each joint. This is illustrated in figure 4.5. We have chosen to implement damping in joint space, because it also accounts for damping in the null space of a redundant manipulator.

The dynamic equations of a multi-DOF serial manipulator are given by equation (3.28). We can generalize this equation by rewriting it as:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F(q,\dot{q})^T + \frac{\partial V}{\partial q}(q) = \tau_a^T + J^T(q)(W^{0,ee})^T \tag{4.4}$$

Where $F(q,\dot{q}) \in \mathbb{R}^n$ is a co-vector of friction forces, $V(q)$ is the potential energy due to gravity and $W^{0,ee}$ is an external wrench acting on the end effector.

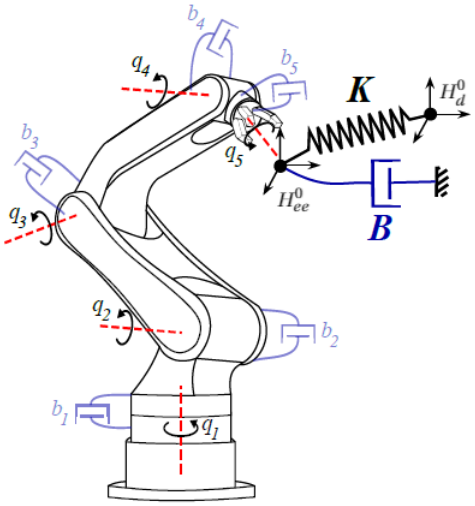


Figure 4.5: Cartesian impedance control of a multi-DOF serial manipulator. A multidimensional virtual spring $K \in \mathbb{R}^{6\times6}$ is used to move the end effector from its current pose H^0_{ee} to its desired pose H^0_d. Damping can be added by a multidimensional damper $B \in \mathbb{R}^{6\times6}$ parallel to the spatial spring or in joint space by a damper b_n on each joint [18].

The spatial spring and damper introduce a potential energy function V_c and Rayleigh function $\mathscr{R}(q)$, which are given by:

$$\begin{aligned} V_c(q) &= -V(q) + \frac{1}{2}(q-q_d)^T K(q-q_d) \\ \mathscr{R}(q) &= \frac{1}{2}\dot{q}^T B(q)\dot{q} \end{aligned} \tag{4.5}$$

Similar to equation (4.3), we can shape the potential energy by:

$$\begin{aligned} \tau_a^T &= -\frac{\partial V_c}{\partial q}(q) - \frac{\partial \mathscr{R}}{\partial \dot{q}} \\ &= -\frac{\partial V_c}{\partial q}(q) - B(q)\dot{q} \end{aligned} \tag{4.6}$$

The first term in equation (4.6) shapes the energy due to the spatial spring and the second term describes dissipation of energy through damping. The virtual spring introduces a wrench $W_k^{0,ee}$ on the end effector, which is called *elastic wrench*. In the one-dimensional case, the spring force is given by $F = k_c(x - x_d)$. For the multi-DOF manipulator, the elastic wrench on the end effector is given by $W_k^{0,ee} = K\delta\chi$ [67]:

$$(W_k^{0,ee})^T = \begin{pmatrix} (\tau_k^{ee})^T \\ (F_k^{ee})^T \end{pmatrix} = \begin{pmatrix} K_o & K_c \\ K_c^T & K_t \end{pmatrix} \begin{pmatrix} \delta\theta_{ee}^d \\ \delta p_{ee}^d \end{pmatrix} \tag{4.7}$$

Where $\delta\chi$ is an infinitesimal twist displacement and $K_o,\ K_t,\ K_c \in \mathbb{R}^{3\times3}$ are rotational, translational and coupling symmetric stiffness matrices, respectively. The diagonal terms in K_o and K_t are the direct-related stiffnesses, which determine the stiffness along all 6 DOFs. The off-diagonal terms and the matrices K_c and K_c^T are the coupling stiffnesses between two different degrees of freedom.

The matrix form of the torque $\tilde{\tau}_k^{ee}$ and force $\tilde{F}_k^{ee}$ can be calculated as:

$$\begin{aligned} \tilde{\tau}_k^{ee} &= -2as(G_o R_{ee}^d) - as(G_t R_d^{ee} \tilde{p}_{ee}^d \tilde{p}_{ee}^d R_{ee}^d) - 2as(G_c \tilde{p}_{ee}^d R_{ee}^d) \\ \tilde{F}_k^{ee} &= -R_d^{ee}\, as(G_t \tilde{p}_{ee}^d) R_{ee}^d - as(G_t R_d^{ee} \tilde{p}_{ee}^d R_{ee}^d) - 2as(G_c R_{ee}^d) \end{aligned} \tag{4.8}$$

A proof of this equation is provided in [67]. The matrices $G_o,\ G_c,\ G_t \in \mathbb{R}^{3\times3}$ are co-stiffness matrices, which are defined as:

$$G_x = \frac{1}{2} tr(K_x) I - K_x \quad for \quad x = t, o, c \tag{4.9}$$

The $tr()$ operator takes the sum of the diagonal elements and the operator $as()$ gives the anti-symmetric part of a square matrix:

$$as(M) = \frac{1}{2}(M - M^T) \tag{4.10}$$

Equation (4.8) is a function of the rotation matrix R_{ee}^d and the matrix form of the translational vector p_{ee}^d (and their inverses). When the desired pose is defined with respect to the base frame as H_d^0, these components can be obtained by:

$$H_{ee}^d = (H_d^0)^{-1} H_{ee}^0 = \begin{pmatrix} R_{ee}^d & p_{ee}^d \\ 0_3^T & 1 \end{pmatrix} \tag{4.11}$$

The total dynamic equations of the controlled system are given by [69]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F(q,\dot{q})^T + G(q)^T = \underbrace{J^T(q)(W_k^{0,ee})^T}_{elasticwrench} - B(q)\dot{q} + \underbrace{J^T(q)(W^{0,ee})^T}_{externalwrench} \tag{4.12}$$

With $G(q)$ the potential energy of the manipulator due to gravity. The control law of equation (4.6) can be extended to:

$$\tau_a^T = J^T(q)(W_k^{0,ee})^T - B(q)\dot{q} + J^T(q)(W^{0,ee})^T - \hat{C}(q,\dot{q})\dot{q} - \hat{F}(q,\dot{q})^T - \hat{G}(q)^T \tag{4.13}$$

Where $\hat{C}(q,\dot{q})\dot{q}$ is a compensation term for Coriolis and centrifugal forces, $\hat{F}(q,\dot{q})$ is a compensation term for friction forces and $\hat{G}(q)$ is a compensation term for gravity forces.

We have now described the implementation of a Cartesian impedance controller for a multidimensional serial manipulator, which gives the manipulator a certain compliance. However, several factors - such as a high external forces or low damping - can bring the manipulator in a state which is potentially dangerous for the patient. We will implement safety limitations to prevent the manipulator from reaching such states, which will be further described in the next section.

4.3 Safety-Aware Impedance Control

Safety is a very important issue in autonomous ultrasound imaging, because it should be avoided that the robot reaches a state that can be potentially dangerous for its environment. Also, the body of the robot has to guarantee compliant behavior during possible collisions with the environment. A safety-aware impedance controller was proposed in [63] to establish safe human-robot interaction in cobots. Energy and power limitation are used as safety metrics, because they directly limit the total energy of the system and the power flowing from the controller to the manipulator. These metrics fit well within the energy-based approach of impedance control. We will first describe the implementation of safety-aware control for the simple impedance controller and we will then extend this to the multidimensional Cartesian impedance controller.

4.3.1 Safety-Aware Simple Impedance Controller

Let us consider the system of figure 4.3 with $m_c << m$ and $k_c >> k$ and controller law given by equation (4.3). The total energy of the system is given by the sum of potential and kinetic energy:

$$E_{tot} = \frac{1}{2}k(x_d - x)^2 + \frac{1}{2}m\dot{x}^2 \tag{4.14}$$

The energy can be limited by adjusting the stiffness k when the total energy exceeds a value E_{max} [63]:

$$k = \begin{cases} k_0 & for \quad E_{tot} \leq E_{max} \\ 2E_{max} - \frac{m\dot{x}^2}{(x_d - x)^2} & for \quad E_{tot} > E_{max} \end{cases} \tag{4.15}$$

Where k_0 is the initial stiffness. The controller adjusts the stiffness of the spatial spring in order to make the robot more compliant to compensate for externally supplied energy due to e.g. an accidental collision.

The power P_c that flows from the controller to the manipulator is given by:

$$P_c = \big(k(x_d - x) - b\dot{x}\big)\dot{x} \tag{4.16}$$

After the stiffness parameter is calculated with equation (4.15), the power can be limited by adjusting the damping parameter b when the power exceeds a value $P_{c_{max}}$ [63]:

$$b = \begin{cases} b_0 & for \quad P_c \leq P_{c_{max}} \\ \frac{k(x_d - x)\dot{x} - P_{c_{max}}}{\dot{x}^2} & for \quad P_c > P_{c_{max}} \end{cases} \tag{4.17}$$

Where b_0 is the initial damping. The amount of power that the manipulator can transfer in an unexpected collision with its environment is in this way limited by regulating the power flow from the actuators to the manipulator. Effectively, this means that the controller adds more damping to decrease the effect of the controller torques.

4.3.2 Safety-Aware Impedance Control of Multidimensional Manipulators

Consider a multidimensional manipulator - like the one illustrated in figure 4.5 - with damping injected in joint space. The control law of this manipulator is given by equation (4.13), where $B(q) = diag([b_1 ... b_n])$. The total energy of the manipulator E_{tot} consists of the kinetic energy $T(q, \dot{q})$ and the potential energy due to the spatial spring $V(R^d_{ee}, p^d_{ee})$. The kinetic energy of link i is given by:

$$T_i(q, \dot{q}) = \frac{1}{2} \dot{q}^T M_i(q) \dot{q} \tag{4.18}$$

Where $M_i(q) \in \mathbb{R}^{n \times n}$ is the inertia matrix of link i, which is given by:

$$M_i(q) = J_i^T(q) Ad^T_{H_0^i} I^i Ad_{H_0^i} J_i(q) \tag{4.19}$$

With:

$$J_i(q) = (T_1 \quad T_2 \ ... \ T_i \quad 0 \ ... \ 0) \tag{4.20}$$

I^i is the inertia tensor given by equation (3.21) and twist T_i according to equation (3.17). Note that T_i is used both for the kinetic energy of link i and for column i of the manipulator Jacobian. The total kinetic energy of the manipulator $T(q, \dot{q})$ can be calculated from:

$$M(q) = \sum_{i=1}^{n} M_i(q); \quad T(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} \tag{4.21}$$

The potential energy stored in the spatial spring can be calculated by [67]:

$$V(R^d_{ee}, p^d_{ee}) = V_t(R^d_{ee}, p^d_{ee}) + V_o(R^d_{ee}) + V_c(R^d_{ee}, p^d_{ee}) \tag{4.22}$$

Where R^d_{ee} and p^d_{ee} are the rotation matrix and translational vector from the end effector to the desired configuration. V_t, V_o and V_c are respectively the translational, rotational and coupling components of the potential energy, which are given by [67]:

$$\begin{aligned} V_t(R^d_{ee}, p^d_{ee}) &= -\frac{1}{4} tr(\tilde{p}^d_{ee} G_t \tilde{p}^d_{ee}) - \frac{1}{4} tr(\tilde{p}^d_{ee} R^d_{ee} G_t R^{ee}_d \tilde{p}^d_{ee} \\ V_o(R^d_{ee}) &= -tr(G_o R^d_{ee}) \\ V_c(R^d_{ee}, p^d_{ee}) &= tr(G_c R^{ee}_d \tilde{p}^d_{ee}) \end{aligned} \tag{4.23}$$

The total energy of the manipulator is given by:

$$E_{tot} = T(q, \dot{q}) + V(R^d_{ee}, p^d_{ee}) \tag{4.24}$$

Energy limitation can only directly be applied by limiting the potential energy. From equation (4.23) follows that the potential energy is proportional to the co-stiffness matrices, so the total energy of the manipulator can be scaled by the co-stiffness matrices. Consider an initial set of co-stiffness matrices G_x calculated by equation (4.9). A scaling factor λ can be defined which scales the co-stiffness matrices in order to limit the total energy of the manipulator [63]:

$$G_x = \lambda G_x \tag{4.25}$$

Where the scaling factor λ is given by:

$$\lambda = \begin{cases} 1 & for \quad E_{tot} \leq E_{max} \\ \frac{E_{max} - T(q,\dot{q})}{V(R^d_{ee}, p^d_{ee})} & for \quad E_{tot} > E_{max} \end{cases} \tag{4.26}$$

So the total energy of the controlled system, which is always less than or equal to the maximum energy E_{max}, is given by:

$$E_{tot} = T(q,\dot{q}) + \lambda V(R^d_{ee}, p^d_{ee}) \tag{4.27}$$

The power that is transferred from the controller P_c is given by the sum of the power that flows from the impedance controller to the manipulator P_m and the power consumed for gravity compensation P_g [63]:

$$P_c = \underbrace{(J^T(q)(W^{0,ee}_{K^*})^T - B_i\dot{q})^T\dot{q}}_{P_m} + \underbrace{\hat{G}(q)\dot{q}}_{P_g} \tag{4.28}$$

Where $W^{0,ee}_{K^*}$ is the wrench due to the spatial spring after energy limitation. B_i is the initial joint damping matrix, which can be scaled by a scaling parameter β to limit the power as:

$$B = \beta B_i \tag{4.29}$$

Where the scaling parameter β is given by [63]:

$$\beta = \begin{cases} 1 & for \quad P_c \leq P_{max} \\ \frac{(J^T(q)(W^{0,ee}_{K^*})^T\dot{q} - P_{max}}{\dot{q}^T B_i \dot{q}} & for \quad P_c > P_{max} \end{cases} \tag{4.30}$$

So the power of the controlled system, which is always less than or equal to the maximum power P_{max}, is given by:

$$P_c = \left(J^T(q)(W^{ee,0})^T - \beta B_i\dot{q}\right)^T\dot{q} + \hat{G}(q)\dot{q} \tag{4.31}$$

In this section we expanded the impedance controller with safety limitations in order to establish safe human-robot interaction. For a serial manipulator, this was implemented by equation (4.26) and (4.30) which results in an impedance controller with variable stiffness and damping. However, an instantaneous change in the controller parameters changes the total energy, which affects the passivity of the system. Passivity is an important property for stability of the system and can be resolved by the concept of energy tanks. The concept of passivity and the implementation of energy tanks will be described in chapter 5.

5 | Passivity of Robotic Manipulators

In this chapter we will discuss the concept of passivity for robotic manipulators. First, a short introduction on port-Hamiltonian systems is given, which is a suitable approach for describing the interaction and energy exchange between systems. Subsequently, we will define passivity and explain the Intrinsically Passive Control (IPC) framework, which is one of the fundamental ideas behind our control strategy. We concluded the previous chapter with the notion that an instantaneous change of the controller parameters - due to implementation of the safety limitations - can affect the passivity of the system. This can be resolved by adding energy tanks to the system.

5.1 Control by Interconnection

As mentioned in chapter 4.1, the dynamical interaction between a robot and its environment is characterized by the exchange of energy between them. Let us consider two interacting systems Σ_1 and Σ_2 (figure 5.1). The interaction between the two systems results in an energy flow from Σ_1 to Σ_2, or from Σ_2 to Σ_1, through an interface that is called *power port*. A power port P quantifies the energy flow by means of a linear map [67]:

$$P = \mathcal{V} \times \mathcal{V}^* \tag{5.1}$$

With vector space $\mathcal{V}$ and its dual space $\mathcal{V}^*$. An element in $\mathcal{V}$ is called a *flow (f)* and an element in $\mathcal{V}^*$ is called an *effort (e)*, such that $(f, e) \in \mathcal{V} \times \mathcal{V}^*$. The direction of the half arrow in figure 5.1 represents the positive direction of power flow. In terms of screw theory, a twist is a flow and a wrench is an effort.

Power ports are *power-continous*, which means that power is always conserved in accordance with the first law of thermodynamics. Moreover, power ports are *bi-directional* which means that energy and physical variables can go in both directions. If effort goes from Σ_1 to Σ_2, then flow goes from Σ_2 to Σ_1 and vice versa.

Let us assume now that Σ_1 is a controller that controls manipulator Σ_2. The connection between controller and manipulator is power-continuous, so there is a direct control over the energy that the controller supplies to the manipulator. Passivity, which we will discuss in more detail in the next section, is now inherent in the controller design. This approach of control is called *control by interconnection* [72].

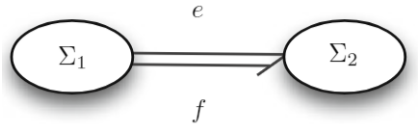


Figure 5.1: Representation of a power port that interconnects system Σ_1 and system Σ_2 [68].

5.2 Port-Hamiltonian Systems

Hamiltonian mechanics are, together with Newtonian mechanics and Lagrangian mechanics, considered as one of the main formalisms of classical mechanics. Hamiltonian mechanics are described by a scalar function $H(p,q)$ called Hamiltonian, that represents the total energy of the system. The variables p and q are n-dimensional vectors of generalized momenta and generalized coordinates respectively. Hamiltonian mechanics are well-suited for port-based modeling and control, because the total internal energy of the system $H(p,q)$ is explicitly expressed in the equations. The Hamiltonian equations are given by [29]:

$$\begin{aligned} \frac{dp}{dt} &= -\frac{\partial \mathscr{H}}{\partial q} \\ \frac{dq}{dt} &= +\frac{\partial \mathscr{H}}{\partial p} \\ \mathscr{H} &= T + V \end{aligned} \tag{5.2}$$

Where T and V are respectively the kinetic and potential energy of the system. This energy-based formulation of physical systems, combined with the concept of power ports, is called port-Hamiltonian system theory.

Let us first assume a system without any dissipation of energy. The total internal energy $H(p,q)$ of this system is constant, while energy is transferred through power ports between interconnected elements. The physical system is controlled by a controller through controller port $\mathscr{C}$. The power-conserving interconnection between these port-Hamiltonian systems is again a port-Hamiltonian system [77]. The overall system can be described in terms of energy exchange by the following equations [29]:

$$\begin{aligned} \dot{x} &= J(x)\frac{\partial H}{\partial x} + g(x)u \\ y &= g^T(x)\frac{\partial H}{\partial x} \end{aligned} \tag{5.3}$$

Where (u,y) is the input-output pair of the controller port $\mathscr{C}$, x is a collection of all states like p and q, $J(x)$ is a skew-symmetric, and consequently power-continuous, internal interconnection matrix and $g(x)$ is a matrix that defines how external power is distributed into the system.

The input-output pair of a port is defined by the power conjugated variables effort and flow. The power conjugated variables of relevant physical domains are given in table 5.1. The port-Hamiltonian description provides a framework for modeling of physical systems that describes the concepts of energy storage and external interaction. We will use this framework in chapter 7 to build a port-based model of the robotic manipulator.

Table 5.1: Efforts and flows of relevant physical domains [62].

Domain	**Effort**	**Flow**
Mechanics (translational)	Force F	Velocity v
Mechanics (rotanional)	Torque τ	Angular Velocity ω
Electric	Voltage U	Current I

5.3 Passivity

Passive systems are a class of dynamical systems which cannot deliver more energy than what is stored [16]. A physical system is passive if the following inequality holds [79]:

$$S(x(t_1)) \leq S(x(t_0)) + \int_{t_0}^{t_1} (w(u(t), y(t))dt \tag{5.4}$$

Where $x(t)$ is the state of the dynamical system at time t, (u, y) is the input-output pair of the connected port, $S(x)$ is a non-negative storage function of energy and $w(u, y)$ is a real-valued function that represents the supply rate of energy. Equation (5.4) is known as the dissipative inequality. Systems for which the inequality holds are called dissipative, which means that there is a irreversible transformation from energy into heat. Systems for which the equality holds are called conservative.

The following theorem, which was first presented by [35] and is valid for any linear and non-linear system, shows the necessity to control a passive environment by a passively-controlled robot:

"Given any non-passive system Σ *with input-output pair (u,y), then there always exists a passive system* $\tilde{\Sigma}$ *that, when connected to* Σ *will give rise to unbounded behavior of the interconnection of* Σ *and* $\tilde{\Sigma}$*."*

The proof of this theorem is given in [29]. From this, it can be concluded that a necessary condition for having stable interaction with an unknown environment is that the controlled robot should behave passively, as seen from the port which interacts with the environment. In the next section we will introduce the IPC framework, which always guarantees that this condition is met.

5.4 Intrinsically Passive Control

The IPC framework basically consists of two parts: the IPC controller and the Supervisor. The IPC controller has a control port connected to the physical robot and another port connected to the supervisor, as illustrated in figure 5.2. The Supervisor takes care of planning, scheduling and coordination of the task to be performed [67]. In analogy with a human, the IPC controller are the muscles and the Supervisor is the brain. The Supervisor is hierarchically speaking a higher level of control, since it supplies energy to the IPC controller. The physical robot has an interaction port connected to the environment and a control port connected to the IPC controller. The IPC controller can supply energy to the robot and drain energy from the robot via actuators. However, when the Supervisor does not supply any energy to the IPC controller, there is no energy available for the IPC controller to move the robot.

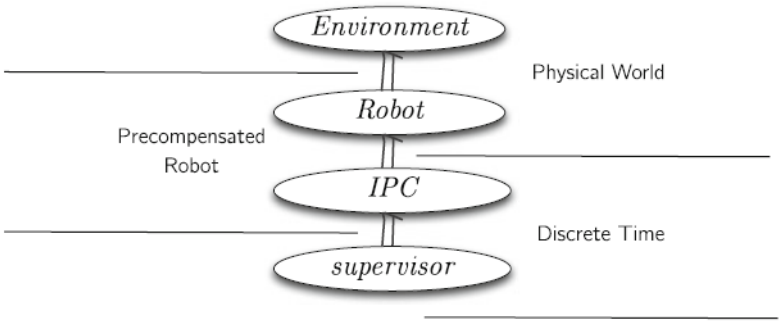


Figure 5.2: IPC architecture [35].

The IPC paradigm states that a stable interaction of the robot with a passive environment can only be achieved when the control is done via interconnection and the control should be passive by itself through the implementation of a Supervisor [68]. A stable interaction between the patient and the manipulator is very important in autonomous ultrasound imaging. First of all because of performance and robustness reasons, but even more because of safety of the patient.

5.5 Energy Tanks

It was noted in chapter 4.3 that the implementation of energy and power based safety metrics affects the passivity of the controlled system, because an instantaneous change of the controller parameters results in internal energy production. The concept of energy tanks, which was proposed by [73], will be used to regain the passivity of the overall system. We will first present the concept of energy tanks for the simple impedance controller and extend this for the multidimensional manipulator.

5.5.1 Simple Impedance Controller with Energy Tank

The concept of energy tanks is fully based on energy flow and energy storage. For this reason, port-Hamiltonian system theory provides an appropriate framework to implement the energy tanks on the controlled manipulator. Let us consider a simple mass-spring system as illustrated in figure 5.3.

The total energy of this system can be written as a Hamiltonian:

$$H(p,x) = \frac{p^2}{2m} + \frac{1}{2}kx^2 \tag{5.5}$$

The dynamics of this system, with state $\underline{\mathbf{x}} = (x,p)^T$ and input-output pair $(F_{ext}, \dot{x})$, can be obtained from equation (5.3):

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{p} \end{pmatrix} &= \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} kx \\ \frac{p}{m} \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} F_{ext} \\ \dot{x} &= \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} kx \\ \frac{p}{m} \end{pmatrix} \end{aligned} \tag{5.6}$$

In order to regain the passivity of the impedance controller with variable stiffness and damping, the controlled system has to satisfy equation (5.4). This can be realized by limiting the energy that is available for the controller through an energy tank. We just discussed the energy and dynamics of a simple mass-spring system. The energy tank can be thought of as a spring, such that the controller (which is only the spring in this case) can only apply forces to the mass when energy is stored in the spring. In order to control the power flow between the energy tank and the mass, the system can be extended with a transmission element MT and computational unit CU [73]. This is illustrated in figure 5.4, with energy tank $H(s)$, transmission ratio u and desired position of the mass x_d.

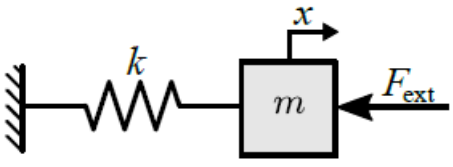


Figure 5.3: Mass-spring system [18].

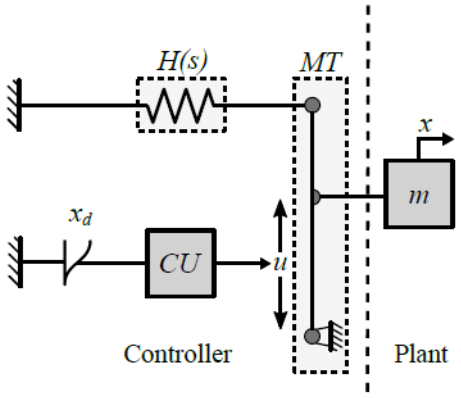


Figure 5.4: Implementation of energy tank for the simple impedance controller [73].

The computational unit CU calculates the variable transmission ratio u. The transmission element MT allows power flow from the controller to the mass under the condition that there is enough energy stored in the tank $H(s)$. The total energy of this system can be written as a Hamiltonian:

$$H(p,s) = \frac{p^2}{2m} + \frac{1}{2}k_{H(s)}s^2 \tag{5.7}$$

When a spring with stiffness $k_{H(s)} = 1$ is used for the energy tank, the dynamics of this system with state $\underline{\mathbf{x}} = (s,p)^T$ are given by:

$$\begin{pmatrix} \dot{s} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & u \\ -u & 0 \end{pmatrix} \begin{pmatrix} s \\ \frac{p}{m} \end{pmatrix} \tag{5.8}$$

Where s is the state of the energy tank $H(s)$. From Newton's second law we know that $\dot{p} = F$, so the bottom part of equation (5.8) can be written as:

$$u_d = \frac{-F_c}{s} \tag{5.9}$$

Where u_d is the desired transmission ratio to obtain controller force F_c in energy tank state s. The controller force results from the variable impedance controller and can be calculated from equation (4.3) and (4.15). The energy tank state can be calculated by integrating $\dot{s} = u\dot{x}$.

It is important that absolutely no force is applied to the plant when the energy tank is depleted or when an external force subtracts energy from the system. This can be realized by modulating the transmission ratio u as:

$$u = \begin{cases} \frac{-F_c}{s} & for(H(s) > \epsilon \wedge P_c < 0) \\ 0 & otherwise \end{cases} \tag{5.10}$$

Where ϵ is a parameter that defines the minimum required energy in the tank, which is given by $H(s) = \frac{1}{2}s^2$. P_c is the power flowing from the controller to the plant, which can be calculated by $P_c = F_c\dot{x}$. P_c will be positive when an external force subtracts energy from the system.

By modulating u in the computational unit CU according to equation (5.10), it is guaranteed that equation (5.4) is satisfied. The internal energy production as a result of the energy and

power limitation is now drained from the energy tank, which is incorporated in the term $S(x(t_0))$. This means that the system with the energy tank included cannot produce internal energy and is therefore passive.

5.5.2 Multidimensional Manipulator with Energy Tanks

The concept of energy tanks can be applied to a multidimensional manipulator by adding an energy tank to every joint, like illustrated in figure 5.5. A computational unit CU computes the transmissio ratio u_i for every joint.

Similar to equation (5.8), the dynamics of a joint i can be described by:

$$\begin{pmatrix} \dot{s}_i \\ \tau_{i,out} \end{pmatrix} = \begin{pmatrix} 0 & u_i \\ -u_i & 0 \end{pmatrix} \begin{pmatrix} s_i \\ \dot{q}_i \end{pmatrix} \tag{5.11}$$

Where $\dot{q}_i$ is the joint velocity and u_i the transmission ratio of tank i. When $P_c < 0$, energy is flowing from the robot to the controller due to e.g. energy transfer between the different links. When equation (5.11) is used to calculate the dynamics of the energy tanks, an energy tank can be fed by energy from other tanks when $P_c < 0$. In that case, it can happen that an energy tank contains more energy than in its initial state. This violates the concept of passivity and hence another condition is added to equation (5.11):

$$if\ (p_{c_i} < 0)\ \ then\ \ \dot{s}_i = 0 \tag{5.12}$$

With $p_{c_i} = \tau_i \dot{q}_i$. The transmission ratio u_i can be calculated by:

$$u_i = \begin{cases} \frac{-\tau_{c_i}}{s_i} & for H_i(s_i) > \epsilon \\ \frac{-\tau_{c_i}}{\gamma^2} s_i & otherwise \end{cases} \tag{5.13}$$

With $\gamma = \sqrt{2\epsilon}$ and τ_{c_i} the resulting torque of the energy tank based controller. Once the transmission ratio is calculated, the output torque to joint i can be calculated by:

$$\tau_{i,out} = -u_i s_i = \begin{cases} \tau_{c_i} & for H_i(s_i) > \epsilon \\ \frac{\tau_{c_i}}{2\epsilon} s_i^2 & otherwise \end{cases} \tag{5.14}$$

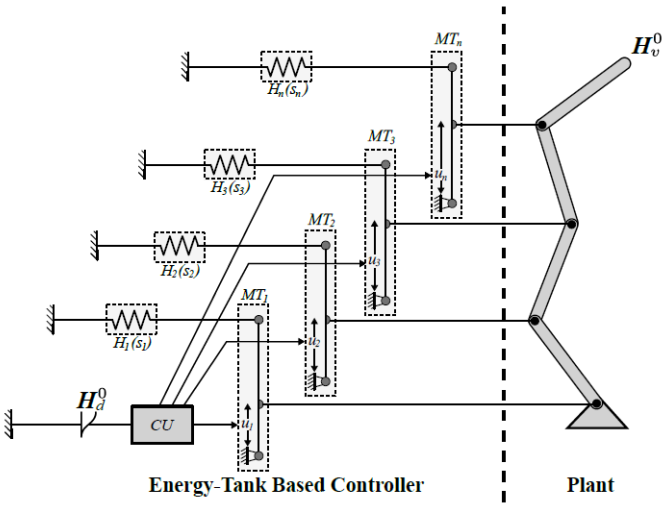


Figure 5.5: Concept of energy tanks for a multidimensional manipulator [18].

The threshold value ϵ is used to indicate when tank $H(s)$ is almost empty. When this happens, the power flow from the controller to the system is limited by adjusting u. In the one-dimensional case of equation (5.10), u is set to zero when the tank is almost empty. In this way, isolation between the controller and the plant is accomplished and the system can only be re-established when an external energy is added to the system. This induces passivity of the system and satisfies the *if*-condition in the upper equation of (5.10).

In the multidimensional case, u is not set to zero and the inequality $P_c < 0$ is removed from the *if*-condition, because the links are connected and energy exchange takes place among consecutive links. When u is set to zero, the tank H_i can be recharged by the compression of the spring in joint i due to the inertial energy of a different link. When the dynamics of u are changed instead of making it zero, like in equation (5.14), the control action still takes place during a depleted state and the previous mentioned problem is circumvented.

The behavior of u, as a function of state s, is given in figure 5.6 for $\tau_{c_i} = -1$. The marked area indicates the region where $H(s) \leq \epsilon$.The main difference is that joint i is gradually shut down in the multidimensional case, where the body is instantly isolated from the controller in the one-dimensional case.

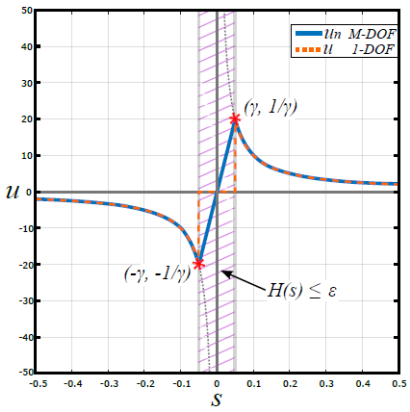


Figure 5.6: Transmission ratio u as a function of tank state s for $F_c = -1$ and $\tau_{c_i} = -1$ [18]. The marked area indicates the region where $H(s) \leq \epsilon$.

5.6 Safety Aware Intrinsically Passive Controller

In the previous sections we discussed safety-aware impedance control, intrinsically passive control and energy tanks. The combination of these concepts into one controller is referred to as the Safety Aware Intrinsically Passive (SAIP) controller. The design of this controller consists of multiple layers, and is illustrated in figure 5.7. The reference block provides the controller with a reference trajectory for the manipulator in the form of desired homogeneous matrices H_d^0. The pose of the end effector H_{ee}^0 can be calculated with direct kinematics from the measured joint positions q. The SAIP controller consists of three layers: a safety layer, a motion layer and a passivity layer. The safety layer works through the equations of section 4.3.2 to inspect the total energy of the system and the power that is flowing from the controller to the robot. In the motion layer, the equation of section 4.2.4 are used in order to calculate the desired joint torques from the homogeneous matrix H_{ee}^d. Finally, the energy tanks are implemented in the passivity layer to ensure the passivity of the overall system.

This concludes the design of the SAIP controller, which consists of an impedance controller with power and energy limitations to increase the safety of the patient and energy tanks to guarantee stability. In chapter 10 and 11, we will implement the SAIP controller in our simulations and experiments in order to test its applicability for autonomous ultrasound scanning. In chapter 6 we will describe an ultrasound-based visual servoing algorithm, which will be added to optimize the ultrasound image quality online.

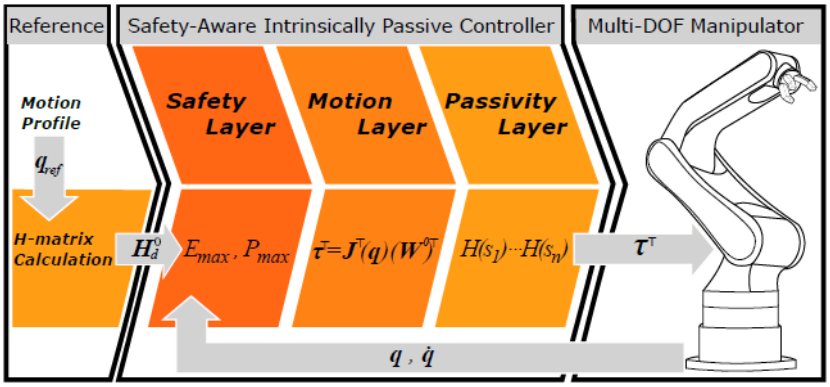


Figure 5.7: Multi-layer design of the SAIP controller [18].

6 | Ultrasound-Based Visual Servoing

One of the main contributions of this project is to include online ultrasound image quality assessment in the control of the robot manipulator. Such a vision-based robot control is referred to as visual servoing. In this chapter we will first explain the concept of ultrasound-based visual servoing after which we will perform a short literature review on ultrasound-based visual servoing algorithms to find an appropriate approach for the task of autonomous ultrasound scanning. Finally, we will explain how this ultrasound-based control can be implemented in addition to our SAIP controller.

6.1 Ultrasound-Based Visual Servoing

Visual servoing techniques are basically a type of robot control where a camera is used as sensor. They can be roughly classified into image-based visual servoing (IBVS), in which information from the images is used to directly control the DOFs of the robot, and position-based visual servoing (PBVS), where a set of 3D parameters is estimated from image measurements to control the robot [21]. In this project we will focus on IBVS, because it allows the camera to search for an optimal trajectory in image space, whereas PBVS allows the camera to follow an optimal trajectory in Cartesian space. The former suits our implementation better, since we want to implement the visual servoing in the probe frame. In the remaining of this section, we will discuss the implementation of a IBVS control law.

Visual servoing schemes aim for minimizing a visual error, which is defined as the difference between a desired feature vector s^* and the current feature vector $s(t)$ [21]:

$$e(t) = s(t) - s^* \tag{6.1}$$

The relation between the twist of the end effector $T_{ee}^{0,0}$ and the variations of s are represented by an interaction matrix $L_s \in \mathbb{R}^{p\times 6}$:

$$\dot{s}(t) = L_s T_{ee}^{0,0} \tag{6.2}$$

Where p is the dimension of the feature vector s. From (6.1) and (6.2) follows that:

$$\dot{e} = L_e T_{ee}^{0,0} \tag{6.3}$$

With $L_e = L_s$. To obtain an exponential decrease of the error, the desired error variation can be defined as:

$$\dot{e}^* = -\lambda e \tag{6.4}$$

Which gives:

$$T_{ee}^{0,0} = -\lambda L_e^+ e \tag{6.5}$$

Where $L_e^+ = (L_e^T L_e)^{-1} L_e^T \in \mathbb{R}^{6\times p}$ is the Moore-Penrose pseudo-inverse of L_e and λ the control gain. These equations can be used in section 6.3 to design a control algorithm for the control strategy that follows from the literature review of section 6.2.

6.2 Ultrasound-Based Control Strategies

This section provides a short literature review on ultrasound-based visual servoing algorithms, with the objective to find an appropriate visual servoing approach to implement in addition to the SAIP controller. Three main approaches to control a manipulator by ultrasound-based visual servoing can be distinguished from literature. In [19] and [20] confidence maps are used, [51] and [34] use an intensity-based visual servoing approach and in [50] and [47] image moments, which are calculated from the pixel intensities, are used. We will discuss these three approaches in more detail in the remaining of this section. In the discussion, we will compare these approaches by grading them on different criteria.

6.2.1 Confidence Maps

Confidence maps were first introduced by [36]. They provide a graphical representation of the per-pixel confidence in the intensities of the image field of view Ω, which emphasizes uncertainties in attenuated and shadow regions. The robot can be controlled to reorient the ultrasound probe in order to improve the confidence in these regions.

By definition, all elements at the beginning of a scan line have a value of 1 (white). They represent the virtual transducer elements. All elements at the end of a scan line have a value of 0 (black). The confidence $C(p)$ of all other pixels $p \in \Omega$ is defined as the probability that an echo, originating from pixel p, reaches one of the transducer elements. An estimation of this probability is performed within the random walk framework [30]. A simplified model of the physics of ultrasound propagation in soft tissues, including attenuation and absorption, is used to calculate the probability that a random walk - starting from a certain pixel - reaches a virtual transducer element. An example of an ultrasound femur image with corresponding confidence map is given in figure 6.1. A shadow region is clearly visible in the bottom right area.

In [19], the in-plane rotation of a convex probe is controlled automatically to improve image quality by using confidence maps. The probe will be oriented such that shadowing objects, such as bones, and attenuated regions are avoided. In this way, it is possible to achieve high confidence on structures beneath shadowing objects. As a visual feature, the angle between the central scan line and the scan line that goes through the confidence-weighted barycenter of the image is used. In [20] two different approaches are presented. In the first approach, force control and confidence control are used to control the tilt and contact force of the convex probe. Translation on the patient's skin and rotation around the depth axis of the probe are teleoperated. In the second approach, image quality is optimized while a target is tracked in the image center. The redundancy framework [64] is implemented to use the redundant DOFs of the manipulator for additional tasks (force control, target tracking), in a hierarchical manner, beside the confidence control task.

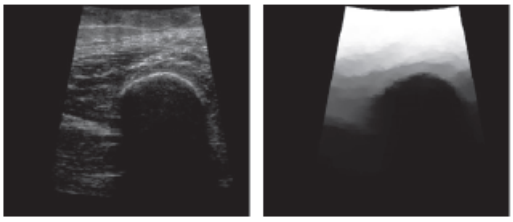


Figure 6.1: Ultrasound femur image (left) [9] and corresponding confidence map (right)

6.2.2 Intensity-based Visual Servoing

Intensity-based visual servoing techniques use the pixel intensities of the ultrasound images as visual features. The pixel intensities correspond to the gray-scale values of an ultrasound image in B-mode, which represent the echogenicity of the corresponding tissue. These techniques do generally not require any image segmentation or a priori knowledge of the target shape and deal very well with the low signal-to-noise ratio (SNR) of ultrasound images.

In [34], an intensity-based image analysis is proposed to determine whether the image acquired by a linear probe is optimal or not. An image is said to be optimal if the root mean square error (RMSE), where error refers to the difference in corresponding pixel values of two consecutive images, exceeds a certain threshold. This value is dependent on the initial image and is determined by a Support Vector Machine (SVM) classifying algorithm. If the image is optimal, it is saved for further diagnosis and the robot moves to the next point on the trajectory. If the image is not optimal, the contact force is increased by a fixed value ΔF and compared to the image that is acquired in the previous position. In [51], a visual servoing approach for intraoperative tracking of the motion of tissue targets and surgical instruments with a 3D probe is proposed. This method takes the intensity values of the voxels of a 3D region of interest (ROI) as visual features. The intensity variation of a voxel along the three orthogonal axes of the ultrasound image frame are used to calculate the interaction matrix, which relates the variation of the pixel intensities to a desired end effector twist.

6.2.3 Moments-based Visual Servoing

Moments-based visual servoing techniques aim for improving the visualization of a certain target by using so-called image moments. Image moments are derived after segmentation of the target and can provide information about the location of the centroid of the target, its orientation and its area.

In [50], a moment-based visual servoing is proposed for automatic positioning of a 3D probe with respect to a target. The in-plane motions in three orthogonal planes are considered to control the probe in 6 DOFs. A 6-dimensional image feature vector is used that contains the coordinates of the mass center of the target cross section in the image plane x_{g0}, y_{g0}, the x-coordinate of the target cross section in the sagittal plane x_{g1} and its main orientation in the three image planes $\alpha_1, \alpha_2, \alpha_3$. These features are illustrated in figure 6.2. This results in a control law in which all elements can be directly measured in the current ultrasound image. In [47], a 2D convex probe is controlled both in-plane and out-of-plane to reach a desired cross-section image of a motionless target. The out-of-plane motions are controlled by an on-line method that uses successive 2D US images to estimate the normal vector. A 6-dimensionsional feature vector is used to control the 6 DOFs, including the the coordinates of the mass center, main orientation and area of the target cross and two moments that are invariant to the image scale, translation and rotation.

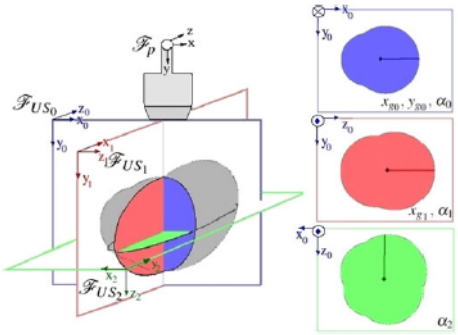


Figure 6.2: Moments-based visual features from [50]. The coordinates of the mass center of the target cross section in image i are given by x_{gi}, y_{gi} and the main orientation in the three image planes is given by α_i.

6.2.4 Discussion

We discussed several approaches for ultrasound-based visual servoing and their application in different studies. To find out which approach is best suited for our application, we will define several criteria on which we can compare the different approaches. In our setup we use a 2D linear probe, because it provides higher resolution and a wider field of view compared to a 3D probe. However, 2D probes only provide a 2D ultrasound image, which makes it more complicated to control out-of-plane motions. The goal of the visual servoing is to adjust the motion of the probe to improve image quality. For a 2D probe, this concerns at least a motion that is normal to the skin surface and an in-plane rotation. The other DOFs will be controlled only by the SAIP controller. It is therefore desirable to be able to control at least these 2 DOFs with the visual servoing algorithm. Visual servoing strategies can be roughly categorized into ones that aim for improving the global image quality and ones that aim for optimizing the visibility of a target. In this project we aim for optimizing the global image quality.

A comparison between the different studies and their approaches is given in table 6.1. We compared the different approaches on probe type, controlled DOFs and whether they aim for improving the global image quality or the visibility of a target. In our setup, we use a linear 2D probe that requires control of at least 2 DOFs for improving global image quality. It follows from table 6.1 that none of the studies that we discussed use an approach that exactly meets our requirements. However, the only difference in the studies on confidence maps ([19] and [20]) is that a convex probe is used instead of a linear probe and the only difference in the intensity-based study [34] is that only 1 DOF can be controlled instead of 2. Since the control by confidence maps can be rewritten to be used with a linear probe, we have chosen to use confidence maps as visual servoing technique for our task.

Table 6.1: Comparison of different studies on ultrasound-based visual servoing

Reference	Approach	Probe type	Controlled DOFs	Global/Target
[19]	Confidence maps	convex	2	Global
[20]	Confidence maps	convex	3/6	Global/Target
[34]	Intensity-based	linear	1	Global
[51]	Intensity-based	3D	6	Target
[50]	Moments-based	3D	6	Target
[47]	Moments-based	convex	6	Target

6.3 Implementation of Ultrasound-Based Control Law

In this section we will describe how confidence maps can be integrated in the design of our control. A predefined trajectory defines a desired pose of the probe over time, with an orientation such that the scan lines travel normal to the skin surface. Confidence maps will be used to adjust the translational motion normal to the skin and the in-plane rotation of the probe online. This means that two DOFs are controlled both by trajectory following as by visual servoing. A weighting factor $w \in [0,1]$ will be used to determine the contribution from trajectory following and the contribution of the control actions resulting from the visual servoing. An additional transformation matrix H_{vs}^{ref} is introduced to change the reference pose H_{ref}^{0} according to the visual servoing algorithm:

$$H_{vs}^{0} = H_{ref}^{0} H_{vs}^{ref} \tag{6.6}$$

$$H_{vs}^{ref} = \begin{pmatrix} cos(w\theta_c) & 0 & -sin(w\theta_c) & 0 \\ 0 & 1 & 0 & 0 \\ sin(w\theta_c) & 0 & cos(w\theta_c) & wdz \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{6.7}$$

With θ_c the in-plane rotation and dz the translation normal to the skin surface that result from the visual servoing algorithm. The transformation matrix H_{vs}^{0} then represents the reference pose of the probe with respect to the base frame, which is a trade-off between trajectory following and visual servoing.

Confidence maps provide a graphical representation of the per-pixel confidence in the intensities of the image. The confidence is calculated by the probability that a reflected ultrasound wave - represented by a random walk originating from a pixel p - reaches one of the transducer elements of the probe [30]. The reflected waves travel through different soft tissues, so a simplified model of the physics of ultrasound propagation in soft tissue is used to calculate the confidence. This model includes Beer-Lambert attenuation, reflection and transmission at tissue boundaries and Ohm's and Kirchoff's law to penalize for a random walk crossing a horizontal or diagonal edge. This method has three free parameters α, β and γ which control respectively the importance of attenuation, absorption and having enough contrast in the confidence map.

The implementation of the confidence maps is illustrated in figure 6.3. Elements at the top row represent the virtual transducer elements, which always have a value of 1 (white). As a boundary condition, all element at the bottom row have a value of 0 (black), representing the end of a scan line. All elements in between represent a single pixel, whose confidence is determined by the weighting factors w_h and w_v. The weighting factor w_v describes the attenuation and absorption between two adjacent pixels p, q along a scan line:

$$w_v(p,q) = exp(-\beta|\hat{I}(p) - \hat{I}(q)|) \tag{6.8}$$

With $\hat{I}(p)$ the attenuated intensity at pixel p, defined by the Beer-Lambert Law:

$$\hat{I}(p) = I(p)exp(-\alpha d(p)) \tag{6.9}$$

And $d(p)$ the depth of p. The weighting factor w_h penalizes horizontal translations in the random walk between two pixels p, q at the same depth and on adjacent scan lines:

$$w_h(p,q) = exp(-\beta(|\hat{I}(p) - \hat{I}(q)| + \gamma||p-q||)) \tag{6.10}$$

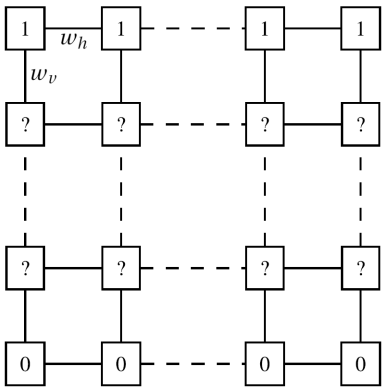


Figure 6.3: Graph setup for the computation of confidence maps using a 4-connected lattice [20].

Once the graph is constructed by calculating the weights, the confidence map can be solved by using the random walks framework. In section 6.1 it is derived how a desired twist can be derived from a visual error. Let us first consider the in-plane rotation, for which the visual error is given by:

$$e_\theta(t) = \theta(t) - \theta^* \tag{6.11}$$

With $\theta(t)$ the current angle between the line that goes through the confidence-weighted barycenter of the image and $\theta^* = 0$ representing the central scan line. The confidence-weighted barycenter $\mu = (\mu_x, \mu_y)$ is a weighted average of all values of the confidence map and can be calculated by [20]:

$$\mu_x = \frac{1}{C_{\Omega_x}} \int\int_{x,z\in\Omega} xC(x,z,t)zdzdx, \quad C_{\Omega_x} = \int\int_{x,z\in\Omega} C(x,z)zdzdx \tag{6.12}$$

$$\mu_z = \frac{1}{C_{\Omega_z}} \int\int_{x,z\in\Omega} zC(x,z,t)xdxdz, \quad C_{\Omega_z} = \int\int_{x,z\in\Omega} C(x,y)xdxdz \tag{6.13}$$

With Ω the field of view of the ultrasound transducer. The field of view of a 2D linear transducer can be represented by the region $\Omega_{2D} = [x_{min}, x_{max}] \times [z_{min}, z_{max}]$, with the z-direction in the direction of the scan lines and the x-direction normal to the scan lines.

An example of an ultrasound image with corresponding confidence map is given in figure 6.4. The vertical red line represents the central scan line and the cross represents the confidence-weighted barycenter (μ_x, μ_y). The goal is to control the probe such that the confidence-weighted barycenter lies on the central scan line ($\mu_x = 0$) and the error $e_\theta(t)$ converges to zero. The control action for in-plane rotation θ_c can be calculated from equation (6.14).

$$\theta_c = arccos\left(\frac{\mu_x}{\mu_z}\right) \tag{6.14}$$

When a specific target has to be localized, it is desirable to have the target in the confidence-weighted barycenter of the image. Now consider the vertical position of the confidence-weighted barycenter. The visual error is given by:

$$e_{\mu_z} = \mu_z(t) - \mu_z^* \tag{6.15}$$

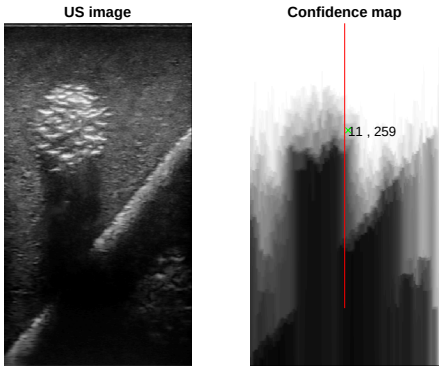


Figure 6.4: Ultrasound image with corresponding confidence map. The vertical red line represents the central scan line and the cross represents the confidence-weighted barycenter.

This error can be reduced by a translation of the probe normal to the skin surface, which results in an increase or decrease of the contact pressure between the probe and the skin. The desired vertical position of the confidence-weighted barycenter μ_z can be placed at a specified height which can be the height of the target. The relation between the translation of the probe and vertical displacement of the confidence-weighted barycenter is dependent on, among others, the structure of the underlying tissue and the current depth of the confidence-weighted barycenter. Experiments will be performed in chapter 11.3 to analyze the relation between translation of the probe normal to the skin surface and vertical displacement of the confidence-weighted barycenter.

In this chapter we introduced the concept of ultrasound-based visual servoing and we selected confidence maps as an appropriate algorithm for ultrasound-based visual servoing. It is described how the control actions can be calculated from the confidence maps. We use a weighting factor to quantify the relative importance between trajectory following and the control actions from the visual servoing algorithm. Calculation of the desired end effector twist $T_{ee}^{0,0}$ from equation (6.5) is performed within the SAIP controller. In this way, we designed a control strategy that incorporates safety limitations, ensures passivity and optimizes the image quality online.

7 Port-Hamiltonian Robotic Manipulator Model

In this chapter, we will derive the dynamic model of an n-link articulated robot manipulator by using port-Hamiltonian system theory. Port-Hamiltonian systems clearly express storage, dissipation and exchange of energy in a system. For that reason, they are well-suited for controller design of physical systems based on passivity theory. Bond graphs are a powerful modeling technique that provide a graphical representation of a physical system, which can be used for modeling and control of port-Hamiltonian systems. Their objective is similar to that of block diagrams, but the main difference is that conservation of energy (the first principle of thermodynamics) is enforced in bond graphs. Energy flows are modeled as power ports, as described in section 5.2. The main rules and elements of bond graphs, as proposed by Paynter in [55], are described in Appendix B. We will use the component-based modeling approach of [25] to build a model of an n-link articulated robot manipulator, which consists of a base, n joints and n links. The model is created with the 20-sim software.

7.1 Base Model

The base of the robot is modeled as the lowest body between the global frame of the robot and the first joint. Let us consider three coordinate frames: Ψ_g representing the global world frame, Ψ_0 representing the global robot frame and Ψ_b representing the unrotated frame at the first joint. These frames are illustrated in figure 7.1, where the red box indicates the base.

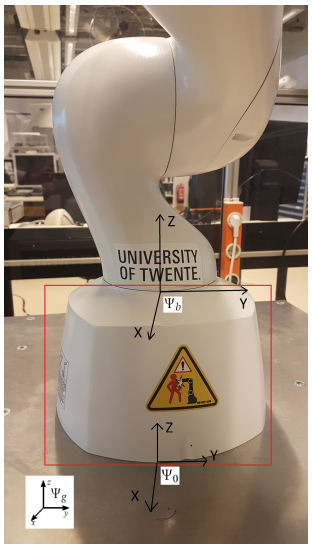


Figure 7.1: Coordinate frames of the base frame. Ψ_g represents the global world frame, Ψ_0 represents the global robot frame and Ψ_b represents the unrotated frame at the first joint. The red box indicates the base.

The bond graph model of the base takes one power port and one homogeneous transformation matrix as inputs and one power port and one homogeneous transformation matrix as output (figure 7.2). The incoming power port is connected to a flow source element S_f with zero flow, because we consider a global robot frame that is stationary with respect to the global world frame. The incoming homogeneous transformation matrix H_0^g is a constant 4-by-4 identity matrix, representing the pose of the global robot frame with respect to the global world frame. The outgoing power port will be connected to the incoming power port of the first joint and the outgoing homogeneous transformation matrix H_b^0 will be an input signal of the first joint.

The graphical model of the base is straightforward, since it only contains a translation in z-direction (figure 7.3). The energy transformation element $\mathbb{TF}$ uses the adjoint matrix $Ad_{H_b^0}$ in equation (3.10) and (3.20) to calculate the dual variables (T, W) of the outgoing power port. It also updates H_b^0 from which H_b^g can be calculated by equation (3.4).

In the remaining of this report we assume that the global robot frame is stationary with respect to the global world frame, which means that H_0^g is a 4-by-4 identity matrix. From now on, we will describe all frames with respect to Ψ_0.

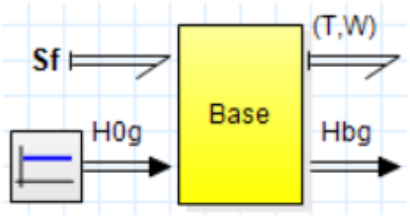


Figure 7.2: Interface model of the base.

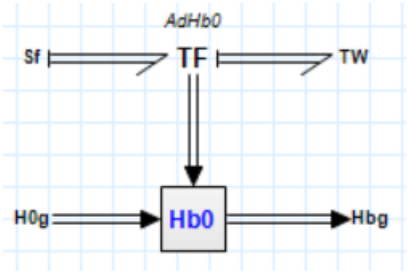


Figure 7.3: Graphical model of the base.

7.2 Joint Model

A joint provides an energetic connection between two connected links. Let us assume robot links with coordinate frames corresponding to figure 7.4. Frame Ψ_j of link n is connected to frame Ψ_i of link *n+1*. A joint enables a one-dimensional rotation of frame Ψ_i of link *n+1* with respect to frame Ψ_j of link n. Ψ_k represents the principal inertial frame.

The bond graph model of a joint takes a power port, a homogeneous transformation matrix and an actuator torque as inputs and has a power port, a homogeneous transformation matrix, the joint position and the joint velocity as output (figure 7.5). The incoming power port is connected to the outgoing power port of the previous link. The incoming homogeneous transformation matrix H_i^0 is connected to the outgoing transformation matrix H_j^0 from the previous link. The outgoing power port will be connected to the incoming power port of the

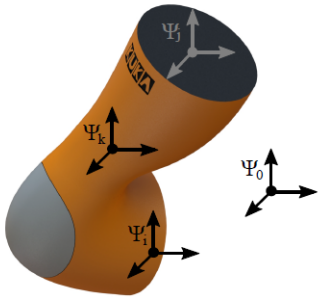


Figure 7.4: Coordinate frames of a robot manipulator link

next link and the outgoing homogeneous transformation matrix H_j^0 will be an input signal of the next link. The other input and output signals will be used as input and output signals of the controller. A modulated effort source $\mathbb{MS}$ is used for the input torque, because it introduces an additional effort that is modulated by the controller.

The graphical model of a joint is illustrated in figure 7.6. The bottom part is similar to that of the base frame, but with a **0**-junction to include the actuator torque and a modified transformer element $\mathbb{MTF}$ because H_j^i is modified by the actuator torque. The effective actuator wrench W^{act} applied by actuator j is equal to the wrench applied to both links:

$$W^{act\ j} = W^{link\ i} = W^{link\ j} \tag{7.1}$$

The twist of link j with respect to frame Ψ_0 is the summation of the twist of link i with respect to Ψ_0 and the twist of link j with respect to Ψ_i:

$$T_j^{j,0} = T_i^{j,0} + T_j^{j,i} \tag{7.2}$$

We use a **0**-junction to include the actuator torque , because the effort (wrench) of all power bonds is equal and the sum of flows (twist) going in the junction is equal to the sum of flows going out.

The effective wrench applied on both links, resulting from actuator torque τ, is given by:

$$W^j = W^i = \hat{W}\tau_j \tag{7.3}$$

With $\hat{W} = [\hat{\tau}, \mathbf{0}]$, which gives a pure torque around the axis of rotation. Power continuity imposes a relation between the the twists and wrenches of the connected links and the actuator torque and angular velocity:

$$W^j T_j^{j,j-1} = \hat{W}\tau_j T_j^{j,j-1} = \tau_j \hat{W} T_j^{j,j-1} = \tau_j \dot{q}_j \tag{7.4}$$

So the relation between angular velocity and twist is given by:

$$\dot{q}_j = \hat{W} T_j^{j,j-1} \tag{7.5}$$

Equation 7.3 and 7.5 describe a linear and multi-dimensional relation between respectively efforts and flows. We consider an articulated robot manipulator with joints that constrain

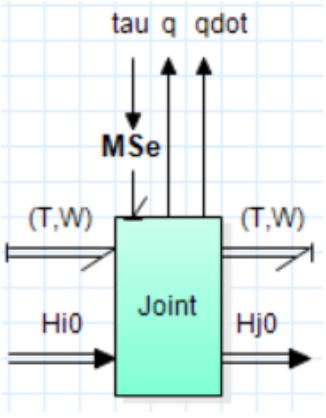


Figure 7.5: Interface model of a joint.

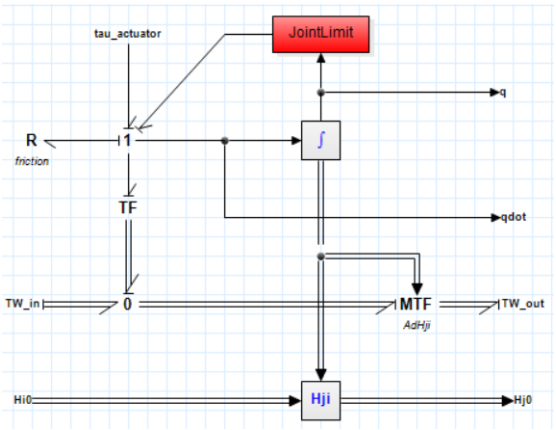


Figure 7.6: Graphical model of a joint.

motion in five DOFs and allow motion around a single axis. This constraint can be modeled by a $\mathbb{C}$ element in integral form together with an $\mathbb{R}$ element. Since they constrain the same flow (twist) they are connected by a **1**-junction. The resistive $\mathbb{R}$ element is constructed by setting the gains to infinity in the constrained directions and setting null gain in the direction that allows motion. For a rotation around the z-axis, this gives $\mathbb{R} = diag([\infty,\infty,0,\infty,\infty,\infty])$. A $\mathbb{C}$ element in integral form is a storage element that takes flow as input and effort as output. By setting high gains on the constrained directions and low gains in the direction where motion is allowed, the DOFs can be constrained. For rotation around the z-axis, this gives $\mathbb{C} = diag([\infty,\infty,-\infty,\infty,\infty,\infty])$. The wrenches that result from the constraint are given by:

$$\begin{aligned}(W_C)^T &= \mathbb{C}\cdot\int T_j^{j,j-1}dt \\ (W_R)^T &= \mathbb{R}\cdot T_j^{j,j-1}\end{aligned} \tag{7.6}$$

This relation is implemented inside the $\mathbb{TF}$ element, which effectively creates a revolute joint. Joint friction is implemented by an $\mathbb{R}$ element. A very basic friction model is used, including only Coulomb friction and viscous friction. This model is illustrated in figure 7.7. Coulomb friction describes the torque that is needed to overcome the static friction. Viscous friction describes how friction increases with increasing joint velocity [5].

The joint velocity $\dot{q}$ is integrated over time to obtain the joint position q. Both the joint velocity and the joint position are used as feedback signals when the controller is implemented. Joint limitations are modeled by the *JointLimit* block, which takes the joint position as input and has a power port as output. The content of the *JointLimit* block is illustrated in figure 7.8. The joint limit is modeled by a virtual wall, represented by a damper with high damping ratio and a spring with low compliance [45]. The damper is implemented by an energy dissipation element $\mathbb{R}$ and the spring by a $\mathbb{C}$ storage element. The *Collision Checker* block outputs a boolean to detect whether the new joint position is outside its range of motion. When this happens, the *if-statement* activates the spring and damper, which effectively limits the rotation of the joint. When the joint position is inside its range of motion, the spring and damper are decoupled and do not influence the motion of the joint. A **1**-junction element is used to connect the power ports of the actuator torque, joint friction, joint limitation and the $\mathbb{TF}$ element, because they all restrict the same motion $\dot{q}$.

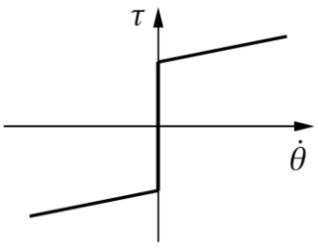


Figure 7.7: Friction model including Coulomb friction and viscous friction [5].

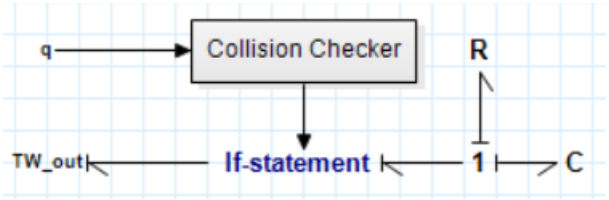


Figure 7.8: Model of joint limitations in a joint

7.3 Link Model

Let us consider a link as illustrated in figure 7.4. The bond graph model of a link takes one power port and one homogeneous transformation matrix as inputs and one power port and one homogeneous transformation matrix as output (figure 7.9). The incoming power port is connected to the outgoing power port of a joint. The incoming homogeneous transformation matrix H_i^0 is connected to the outgoing homogeneous transformation matrix H_{j-1}^0 of the previous link. The outgoing power port will be connected to the incoming power port of the

next joint and the outgoing homogeneous transformation matrix H_j^0 will be an input signal of the next joint.

The graphical model of a link is illustrated in figure 7.10. The bottom part is again similar to that of the base. The links are considered to be rigid bodies with uniform mass distribution. The rigid body dynamics of a link consist of inertial effects, Coriolis and gyroscopic effects, and gravitational effects.

We define a frame Ψ_k as the principal inertia frame. The balance of wrenches that are acting on link j is given by:

$$\mathscr{I}^k \dot{T}_j^{k,0} = \begin{pmatrix} \tilde{\omega}_k^{k,0} & \tilde{v}_k^{k,0} \\ 0 & -\tilde{\omega}_k^{k,0} \end{pmatrix} \mathscr{I}^k T_k^{k,0} + (W^k)^T \tag{7.7}$$

With $I^k = \begin{pmatrix} J_j & 0 \\ 0 & m_j I \end{pmatrix}$ the inertia tensor of body j, expressed in frame Ψ_k. J_j is a diagonal matrix containing the principal inertia and m_j is the mass of link j. $\tilde{\omega}_k^{k,0}$ and $\tilde{v}_k^{k,0}$ are the matrix form of respectively the rotational part and translational part of the twist of frame Ψ_k with respect to frame Ψ_0, expressed in frame Ψ_k. $(W^k)^T$ is the external wrench on link j, expressed in frame Ψ_k [70].

The momentum screw of a rigid body is given by:

$$(\mathscr{P}^{c_j})^T = \mathscr{I}^{c_j} T_{c_j}^{c_j,0} \tag{7.8}$$

Which is comparable to the momentum of a point mass $p = mv$. Similarly, the derivative of the momentum of a point mass $\dot{p} = F$ can be generalized to rigid body dynamics as:

$$\dot{\mathscr{P}}^{0,j} = W^{0,j} \tag{7.9}$$

The first term of equation (7.7) represents the change in momentum screw (assuming that $\mathscr{I}^{c_j}$ is constant), resulting from a change in twist. This can be regarded as the inertial effect, represented by a storage element in differential form, that is a function of the flow variable, i.e. an I element. The second term of equation (7.7) are the Coriolis and gyroscopic effects.

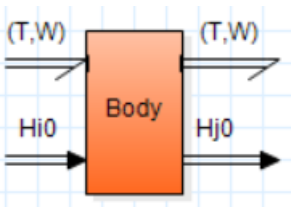


Figure 7.9: Interface model of a link

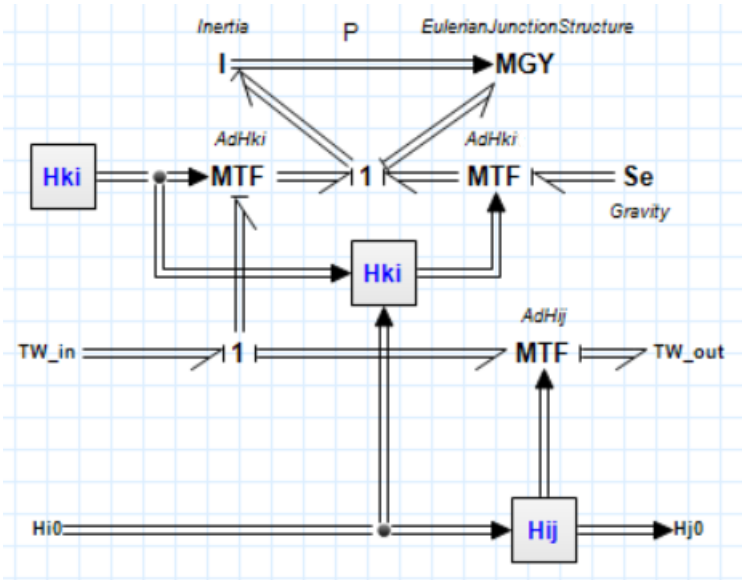


Figure 7.10: Graphical model of a link

It relates an output effort (wrench) with an input flow (twist), so this is represented by a $\mathbb{MGY}$ element that is modulated by the momentum screw. The last term of equation (7.7) corresponds to sum of the wrench that is applied by the joint actuator on the link and the wrench that is introduced by gravity. Since all elements share the same flow (twist), they are connected by a **1**-junction. This is represented by the top part of figure 7.10.

A coordinate transformation is required to relate the twist introduced by the joint to the twist in the principal inertia frame:

$$T_i^{i,0} = Ad_{H_k^i} T_i^{k,0} \tag{7.10}$$

And similar for the coordinate transformation between the wrench expressed in frame Ψ_i and the wrench expressed in frame Ψ_k:

$$(W^k)^T = Ad_{H_k^i}^T (W^i)^T \tag{7.11}$$

The adjoint is represented by an $\mathbb{MTF}$ element, parameterized by $Ad_{H_k^i}$.

The wrench resulting from gravity W_g^i is obtained by coordinate transformation of the wrench W_g^k by the adjoint $Ad_{H_k^i}$. The adjoint is a function of the joint position q_i, so it is represented by an $\mathbb{MTF}$ element. The pose of frame Ψ_k with respect to the base frame Ψ_0 is used to calculate the amount of energy in the effort source S_e due to gravity.

7.4 Manipulator Model

In this chapter we described how joints and links can be modeled in the 20-sim software by using port-Hamiltonian system theory. An n-DOF articulated robot model can be built by connecting the base to n joints and links. This is illustrated in figure 7.11 for a 1-DOF robot. An interaction force is included at the end effector of the robot. A motion profile element is used to simulate a time-dependent interaction force F_{env}, which interacts with the end effector through an effort source. A **1**-junction is used between the end effector and the environment, because the external force restricts the motion of the end effector.

In chapter 10, we will implement the SAIP controller that we designed in chapter 4 in the 20-sim software. The models of the joints and links that are described in this chapter will be used to create a model of the KUKA LBR Med, which can be used to test and validate the controller in simulation. Subsequently, the controller will be converted to C++ code to be implemented on the physical robot.

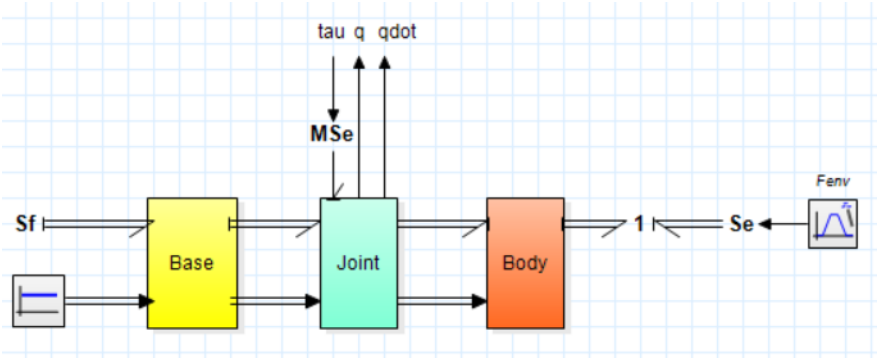


Figure 7.11: Model of a 1-DOF articulated robot

8 Methods

This chapter describes the system specifications, including both hardware and software, as used in the experiments of this project. The design and production of an ultrasound phantom is also included at the end of this chapter.

8.1 Hardware Specifications

The main hardware components that are used in this project are the robotic manipulator and the ultrasound system. The components of both systems will be discussed in this section.

8.1.1 Robotic Manipulator

An overview of the hardware components involved in the robot communication is illustrated in figure 8.1, which consists of the following components:

1. KUKA LBR Med
2. KUKA Robot Controller (KRC)
3. Windows PC
4. Linux PC
5. KUKA SmartPad

We will describe the LBR Med, KRC and SmartPad in more detail below. The role of the Linux PC and Windows PC will be discussed in more detail in the software section (chapter 8.2).

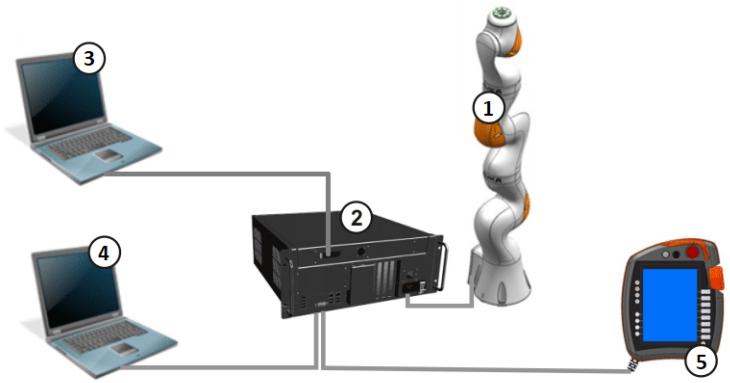


Figure 8.1: System overview for control of the robotic manipulator **(1)** KUKA LBR Med **(2)** KUKA Robot Controller **(3)** Windows PC with KUKA Sunrise Workbench Software **(4)** Linux PC for communication with FRI **(5)** KUKA SmartPad [40].

KUKA LBR Med

The KUKA LBR Med is a medically certified articulated robot with 7 DOFs (figure 8.2). This robot is especially designed for medical applications, such that it meets the medical safety requirements and the processes in the life cycle of medical software development were also verified and approved. The robot has one redundant DOF, which gives it more dexterity and helps avoiding typical singularities of a 6-DOF manipulator. Each joint is equipped with position and torque sensors, such that it can be operated with position control, velocity control and torque control. The range of motion of every joint is given in table 8.1.

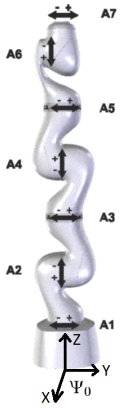


Figure 8.2: Axes of rotation of the KUKA LBR Med.

Table 8.1: Joint limits of the KUKA LBR Med

Range of motion						
Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6	Joint 7
±170	±120	±170	±120	±170	±120	±175

KUKA Robot Controller

The KUKA Robot Controller (KRC) directly controls the robot. Applications created on the Windows PC can be transferred to the KRC through the KUKA Line Interface (KLI). The Linux PC can connect to the KRC through the KUKA Option Network Interface (KONI), which is a UDP-based interface that allows data exchange at a maximum rate of 1 kHz. A human operator can communicate with the robot controller through the KUKA SmartPad, which provides all the operator control and display functions required for controlling the robot [41].

KUKA SmartPad

The KUKA SmartPad allows the user to interact with the robot controller. It can be used for e.g. manually jogging in joint space or world space, information provision of the current robot state, teaching frames to the robot or running Sunrise Workbench applications. The KUKA SmartPad and the KRC can also be accessed through Remote Desktop.

8.1.2 Ultrasound System

The Siemens ACUSON P500 ultrasound system (figure 8.3a) with a L10-5v linear transducer (figure 8.3b) is used in this project for ultrasound imaging. The ACUSON P500 is a portable ultrasound system that facilitates accurate diagnosis and provides an operator-friendly interface. The L10-5v probe has a frequency bandwidth of 5 - 12 MHz and a maximum display depth of 100 mm, which are suitable specifications for breast examinations.

(a) **(b)**

Figure 8.3: (a) Siemens ACUSON P500 ultrasound system [7] and **(b)** L10-5v ultrasound probe [8].

3D-printed Ultrasound Probe

A 3D-printed probe was used in the experimental phase of this project, as well as in the experimental validation of our controller. The 3D-printed probe was attached to the end effector of the robot, as illustrated in figure 8.4a. A media flange inside the end effector of the KUKA LBR Med provides connections for power supply, I/Os or ethernet in order to connect customer specific tools on the flange. In the ideal case, a real ultrasound probe is attached to the end effector in the same way as in figure 8.4a.

3D-printed Probe Holder

During this project we did not have access to an ultrasound probe that can be attached to the media flange of the KUKA LBR Med. Instead, a 3D-printed probe holder was designed to attach the ultrasound probe to the end effector of the robotic manipulator (figure 8.4b). A 3D CAD model of the L10-5V was made from a 3D scan to determine the geometry of the probe holder. The probe is attached to the end effector under an angle of 45, because the wire of the ultrasound probe should not be bent too much. This requires a different homogeneous matrix to describe the midpoint of the contact surface of the ultrasound probe with respect to the end effector of the robotic manipulator.

(a) **(b)**

Figure 8.4: (a) 3D-printed ultrasound probe attached directly to the end effector of the robotic manipulator and **(b)** L10-5V ultrasound probe attached by a probe holder to the robotic manipulator.

Video Capturing Device

The Magewell USB Capture HDMI Gen 2 device [4] was used to transfer images from the ultrasound system to the Linux PC. The confidence estimation can then be performed on the Linux PC and the resulting control actions will be sent to the robot through the FRI. The video capture device has one USB port connected to the linux PC and the other HDMI port connected to the ultrasound system (figure 8.5). The device can handle video resolutions up to 2048x2160 and a frame rate of 120 fps, which is sufficient for our application.

Figure 8.5: Magewell USB Capture HDMI Gen 2 device for transferring ultrasound images from the ultrasound system to the Linux PC.

8.2 Software Architecture

In this section we will describe the software architecture as used in this project. A schematic representation of the total system, including all the software elements, is illustrated in figure 8.6. The Windows PC and Linux PC are both connected to the KRC through ethernet. The Windows PC is used to program applications in Sunrise Workbench, which can be executed on the KRC. The Linux PC can be used for real-time data exchange between a robot application on the KRC and a FRI client application on the Linux PC.

In our approach, the Sunrise application moves the robot to a desired configuration and commands a position hold from there. The FRI client then takes control by sending torque overlays to the KRC. The KRC directly controls the robot, which has the ultrasound probe attached to its end effector. The ultrasound system captures the ultrasound images and sends them to the Linux PC. A Matlab program on the Linux PC estimates the confidence maps, calculates the desired control actions and communicates them to the FRI client. The SAIP controller is developed in 20-sim on the Windows PC and converted to C++ code to be integrated in the FRI. The FRI client includes both the SAIP controller and visual servoing algorithm to control the robot. We will now describe the different software components in more detail.

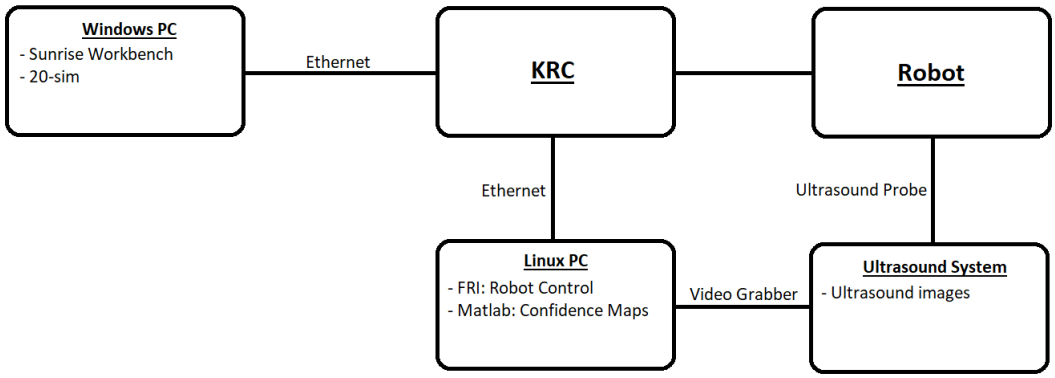


Figure 8.6: Schematic representation of the setup, including all software elements.

8.2.1 Sunrise Workbench

Sunrise Workbench is a tool to program robot applications in Java, which can be executed on the KRC. It allows management of coordinate systems and motions and provides a programming interface for integration of external libraries and functionalities [40]. The robot can be commanded to execute a linear motion, point-to-point motion, circular motion, spline motion or position hold. During these motions, the robot can use position control, axis-specific impedance control and Cartesian impedance control.

8.2.2 Fast Research Interface

The Fast Research Interface (FRI) facilitates continuously and real-time-capable data exchange between a robot application on the KRC and an FRI client application on an external system. A robot application on the KRC that is programmed with Sunrise Workbench, can be overlaid by the FRI with a position, wrench or torque overlay. This allows the user to create C++ applications for real-time control of the robot.

The FRI is a state machine that has four states [40]:

- **MONITORING WAIT:** The KRC has opened the FRI connection and is waiting for real-time-capable data exchange.
- **MONITORING READY:** The KRC is performing real-time-capable data exchange with the FRI client application.
- **COMMANDING WAIT:** The KRC initializes the motion that is commanded by Sunrise Workbench and synchronizes itself with the FRI client.
- **COMMANDING ACTIVE:** The KRC applies the commanded values from the FRI client application for superposing the robot path.

The FRI client cyclically commands position, wrench or torque overlays to the KRC at a maximum send rate of 1000 Hz. A flowchart of the method for cyclical communication is given in figure 8.7. The method receives information about the current state as input. When the FRI state has changed, it calls a callback function to react on the state change. The FRI state machine recognizes the current state of the FRI and cyclically calls the corresponding callback function. When an FRI connection is successfully established, the cl.command() method will be called that includes the robot controller. The robot controller uses the current state of the robot (joint positions and joint velocities) to calculate the desired joint torques. The joint torques are sent to the KRC to move the robot. In this way, we have full control of the robot through the FRI application.

8.2.3 20-sim

The 20-sim software is a modeling and simulation program that allows modeling by equations, block diagrams, physical components and bond graphs [1]. A short overview of the main features of 20-sim is given in Appendix C. We have used the 20-sim software to build a port-Hamiltonian model of the robot, to design our controller and to test the controller in simulation. The controller was then converted to C++ code and implemented in the FRI client application.

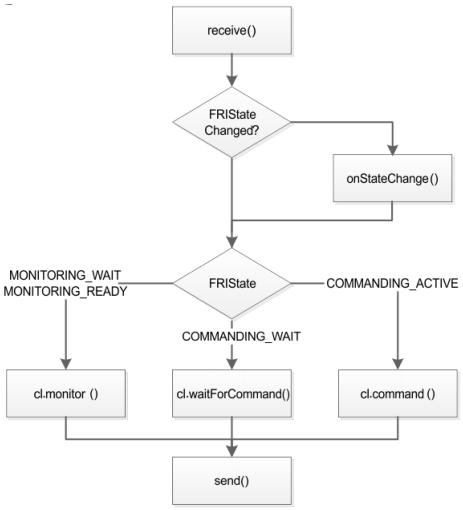


Figure 8.7: Cyclical communication between the FRI client and robot application.

8.3 Ultrasound Phantom

This section describes the design and production of the ultrasound phantom. Phantoms that mimic human body parts are used for experimentation in the MURAB project, because human subjects are not always available and the safety of the human subjects can not always be guaranteed in the experimental phase. Developing a phantom in-house also provides more flexibility on the shape and structure of the phantom compared to using a commercial phantom.

8.3.1 Thigh Phantom

The phantom that is used in this project represents a piece of human thigh. A thigh phantom is used instead of a breast phantom, because the shape and structure of the human thigh is less complex than that of a female breast. Since the system is still in its early development, we decided to first make it work for an easier shape before extending it to more complex shapes. We chose a thigh phantom, because the thigh has a convenient shape for scanning and it contains different muscles and other tissues. We aim to create a thigh phantom that has a similar structure and feeling as a human thigh, and includes different ultrasound phenomena to test the visual servoing algorithm.

8.3.2 Material

The human thigh consists roughly of a layer of skin, fat, muscles and a bone (figure 8.8). The bone will be 3D printed from Polylactic Acid (PLA). The CES EduPack [2] software was used to select an appropriate material for the skin, fat and muscles of the thigh phantom. Density, Young's modulus, tensile strength and castability are used as properties to select the material. The information about human skin and human muscle that is reported in the database of CES EduPack is given in table 8.2. No information about human fat is present in the CES EduPack database. In [43], a density of 0.9 kg/m^3 was reported for human fat.

The materials were filtered according to table 8.2 and a castability in the range of [3,5] on a scale of [0,5]. Castability is an important property of the material, since the material has to be casted into a mold containing different geometries. The selection resulted in three elastomers: Butyl rubber, Polychloroprene and Silicone elastomers. Their properties are illustrated in figure 8.9.

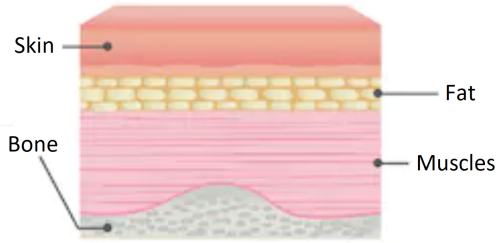
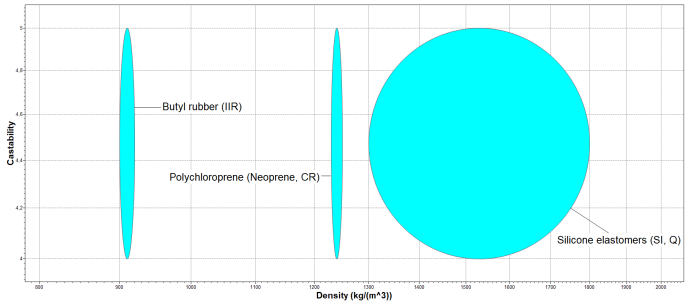


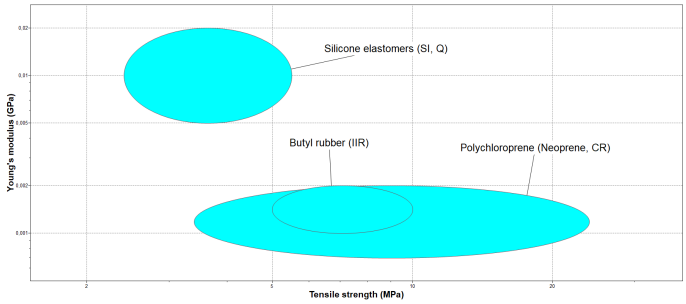
Figure 8.8: Layers of the human skin.

Table 8.2: Material properties of the human skin and human muscle [2].

Material	**Material property**	**Value**
Human skin	Density	ρ = 1110 - 1270 kg/m^3
	Young's modulus	E = 0.001 - 0.004 GPa
	Tensile strength	σ_t = 3 - 9 MPa
Human muscle	Density	ρ = 1300 - 1350 kg/m^3
	Tensile strength	σ_t = 0.1 - 0.3 MPa



(a)



(b)

Figure 8.9: Material properties of Butyl rubber, Polychloroprene and Silicone elastomers **(a)** Density and castability **(b)** Tensile strength and Young's modulus.

It follows that the castability of all three materials is in the same range. The density of Butyl rubber is more like fat, the density of Polychloroprene is like skin and the density of Silicone elastomers is like muscles. From figure 8.9b follows that Silicone elastomers contain the most appropriate combination of the Young's modulus and tensile strength for muscles, while Butyl rubber and Polychloroprene resemble the skin better.

An important property of a phantom is that it yields realistic ultrasound images. The propagation of ultrasound waves in silicon-based materials and rubber-based materials was investigated in [46] and [17] , which yielded satisfying results. In [17], phantoms were made of silicone rubber, which meets the material properties that we defined before. Experiments with

the silicone rubber phantom resulted in realistic ultrasound images. Because silicone rubber has material properties that resemble the human thigh and it yields realistic ultrasound images, we decided to use a silicone rubber for the phantom.

The Ecoflex™00-30 platinum cure silicon rubber was chosen as material for the phantom. It is delivered as two liquid substances, which make a chemical reaction when they are mixed together. Layers of different hardness can be created by adding a Slacker® component. Adding this component makes the product softer and changes the rebound properties, but it also makes the product more sticky. In this way it is possible to mimic the skin layer, fat layer and muscle layer. To every mixture we added about 1 volume percent of silica powder. This gives contrast to the ultrasound images, because of scattering of the ultrasound waves on the silica particles. Silica powder is chosen because these particles are very light and distribute evenly over the substance. If too heavy particles are used, they will sink to the bottom. Air bubbles were removed from the mixture by putting it in a vacuum chamber of 100 millibar underpressure. After that, the liquid can be poured into a mold and it will become solid after 4 hours.

8.3.3 Mold

A mold was designed in Solidworks® and consists of five components such that the casted phantom can easily be removed from the mold. The design of the mold was exported as STL-file and printed from PLA with an Ultimaker 2+ 3D printer [10]. The 3D printed mold, including the bone, is given in figure 8.10. The mold is upside down, so the phantom will have a slight curve (similar to that of the human leg) on top.

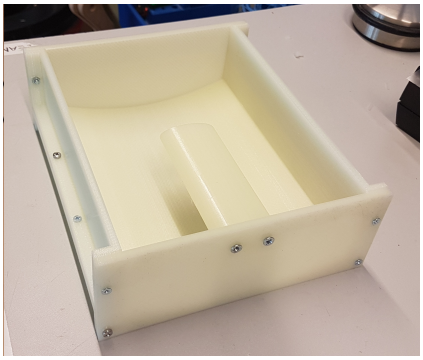


Figure 8.10: 3D printed mold for casting of the phantom.

8.3.4 Structure of the phantom

As mentioned before, the phantom is constructed layer by layer to resemble the human thigh as close as possible. The inner structure of the phantom is illustrated in figure 8.11. By implementing inlays, scattering particles, fish oil capsules and garlic capsules we introduce different ultrasound effects, such as e.g. shadowing. In this way, a phantom is created that includes most of the phenomena encountered in scanning of human body parts.

The first layer (0 - 15 mm) represents the skin and will be in contact with the probe. For this reason, it must stick as little as possible, so we did not add any Slacker®. The second layer (7.5 mm) represents the fat layer, which has to be more soft than the skin and the muscles. We added Slacker® to the silicone rubber for this layer in the ratio of 4:1. We also inserted a 3D printed inlay in the second layer, which we later removed to cast a mixture of different composition. The third layer (10 mm) does not include any additional elements, but includes a small amount of Slacker® in the ratio 30:1. This layer mimics the connective tissue. The remaining layers represent the different muscles in the thigh and do not include Slacker®. The fourth layer (10 mm) includes a garlic capsule and a fish oil capsule. The fifth layer (7.5 mm),

which is the layer that is illustrated in figure 8.11, includes the scanning particles, another garlic capsule and an inlay that was filled with a mixture of high concentration silica powder. The sixth layer (15 mm) fills the mold.

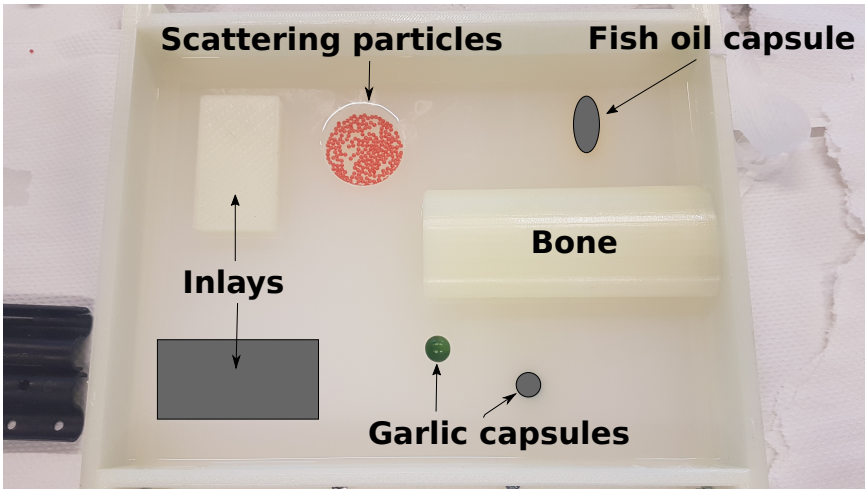


Figure 8.11: Structure of the phantom. Scattering particles, fish oil capsules, garlic capsules and inlays are added to mimic different ultrasound phenomena. Gray elements illustrate elements in underlying layers.

8.3.5 Final Product

An image of the produced thigh phantom is given in figure 8.12. The dimensions of the phantom (length x width x heigth) are (200 x 150 x 65) mm. In chapter 11.2 we will evaluate the quality of the phantom and decide whether we can use it for our experiments that involve ultrasound imaging.

Figure 8.12: Thigh phantom made of silicone rubber.

9 Design of Experiments

In chapter 1.3 we defined two milestones that have to be accomplished in order to meet the research objective. The SAIP controller is first tested in simulation to analyze its behaviour on the port-Hamiltonian robot model that we created in chapter 7. Experiments 1 and 3 are intended to examine both milestones on the physical robot and to validate our proof of concept. Experiment 2 is added to analyze the ultrasound phantom that we produced. The design of these experiments will be discussed in this chapter.

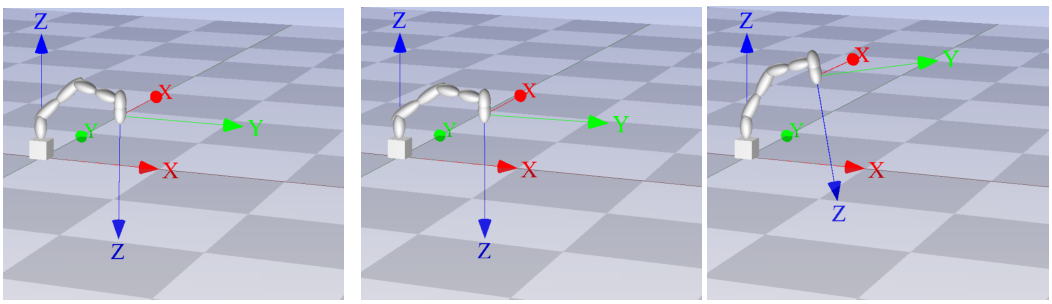
9.1 Simulation Experiment

Simulations are performed to test and validate the SAIP controller before implementation on the physical robot. The goal of the simulation is twofold. First, the simulation environment allows quick testing and monitoring of variables to accelerate the design and development of the controller. Second, testing of the controller in the simulation environment reduces the risk of damage on the physical robot. We will first perform an experiment in which we check whether the joint limitations are correctly implemented in the robot model. Subsequently, we will validate the controller by a simulation experiment that includes the basic motions of an ultrasound scanning task.

A motion is simulated that consists of three phases: *StartPosition, Scanning* and *EndPosition*. The end pose of every phase is illustrated in figure 9.1. The simulation is started from the home position, which is the upright configuration. This trajectory was chosen, because it is very similar to the basic motions of a scanning task with the physical robot. The joint positions in degrees at the end of every phase are as follows:

- StartPosition: $q = (0;45;0;-60;0;75;90)^T$
- Scanning: $q = (0;53;0;-45;0;81;90)^T$
- EndPosition: $q = (0;-27;0;-45;0;97;90)^T$

We will test whether the controller shows desired behaviour regarding trajectory following, safety limitations and joint torques, such that it can be implemented on the physical robot.



(a) **(b)** **(c)**

Figure 9.1: End poses of the three phases of the simulation **(a)** Start of scanning phase **(b)** End of scanning phase **(c)** End pose. The base frame and the end effector frame are illustrated.

9.2 Experiment 1: Compliant Control Strategy

The first milestone of this project is the design and implementation of a compliant control strategy. In chapter 4 and 5 we described the design of the SAIP controller. We designed 4 experiments to test the applicability of the SAIP controller for autonomous ultrasound scanning, which will be explained in more detail below.

9.2.1 Setpoint Experiment

The goal of this experiment is to investigate the behaviour of the robot for the control actions resulting from the confidence maps, i.e. a commanded translation and a commanded rotation. We will examine the influence of the open parameters E_{max}, P_{max}, K_o, K_t, K_c and b on the dynamics of the controller and the robot. From this, we can determine a suitable set of open parameters for the rest of the experiments.

9.2.2 Disturbance Experiment

The goal of this experiment is to investigate how the robot behaves under external disturbances. A fixed setpoint will be used while disturbing the end effector with: a linear force, a torque and a combination of a rotational and translational disturbance. The pose errors, safety limitations and energy components will be monitored during this experiment to examine whether the controller behaves appropriately under the external disturbances.

9.2.3 Energy Tanks Experiment

The goal of this experiment is to demonstrate how the energy tanks get depleted during a task and how the robot behaves when a tank is empty. A block wave on the reference position of the first joint will be commanded to the robot, while keeping the position of the other joints fixed. The first joint has to use energy from the tank to move to the changed reference position. After a while, the energy tank of the first joint will be empty and it should not be able to perform any task. This experiment has to proof the concept of passivity for the SAIP controller.

9.2.4 Scanning Trajectory Experiment

The goal of this experiment is to demonstrate the behaviour of the robot and the controller for trajectory following. For the scanning task it is required to define a scanning trajectory. An STL-file of the phantom, which can be obtained from a 3D CAD model of the mold, is used as starting point for trajectory planning. In case that a scanning trajectory is required for a target volume of the human body, the STL-file can be obtained from the MRI image.

We have chosen for a block-shaped scanning trajectory, because it can cover the total surface of the phantom and it is convenient for the shape of our phantom. Trajectories are created by defining a sequence of poses P_i, which are specified by a six-dimensional vector:

$$P_i = (x_i, y_i, z_i, A_i, B_i, C_i) \tag{9.1}$$

Where (x_i, y_i, z_i) defines the position at step i and (A_i, B_i, C_i) defines the pose in Euler angles with Z-Y-X convention.

A MATLAB script was written to generate trajectories on the phantom. The script is based on the fast, minimum storage ray/triangle intersection (RTI) algorithm [48], which determines whether a ray intersects a triangle. The STL-file describes a triangulated surface of the phantom by vertices and unit normal of the triangles. The RTI algorithm returns a vector (t, u, v) with t the distance to one vertex of the triangle, and u and v the coordinates of the other two vertices with respect to the first one. The value of t can be used to plan the next point of the trajectory,

which is always on the surface of the phantom. The pose is defined by frames which consist of three unit vectors that describe a rotation matrix with respect to a global coordinate frame:

$$R = \left(\hat{x}, \hat{y}, \hat{z}\right) \tag{9.2}$$

The unit vector in z-direction $\hat{z}$ is already available from the STL-file, since it corresponds to the unit normal of a triangle. It was mentioned in chapter 2 that more ultrasound waves will be reflected back to the transducer if the probe is perpendicular to the skin surface, which is beneficial for the quality of the image. For this reason, the orientation of the probe on the predefined trajectory is always such that the probe axis is aligned with the unit normal of the corresponding triangle.

A temporary unit vector in y-direction $\hat{y}_{temp}$ is calculated from two subsequent points and the unit vector in x-direction $\hat{x}$ is then calculated by the cross product between the other two unit vectors:

$$\hat{x} = \hat{y}_{temp} \wedge \hat{z} \tag{9.3}$$

The unit vector in y-direction $\hat{y}$ is then calculated by taking the cross product between $\hat{z}$ and $\hat{x}$:

$$\hat{y} = \hat{z} \wedge \hat{x} \tag{9.4}$$

The block-shaped trajectory is illustrated in figure 9.2. For clarity, the negative z-direction is plotted which is always perpendicular to the surface and pointing out of the phantom. Detailed information about this trajectory is given in figure 9.3. The x-position is increasing step-wise, the y-position is described by a cosine and the z-position is described by a sine. The Euler angles A, B and C are defined corresponding to Z-Y-X convention.

In this experiment we will compare the performance on trajectory following of the SAIP controller with the impedance controller from Sunrise Workbench, which is equivalent to the one described in chapter 4.2.4. Beside, we will also examine the influence of parameter settings on trajectory following for both controllers. Finally, we monitor the safety limitations, energy and joint torques during the scanning procedure.

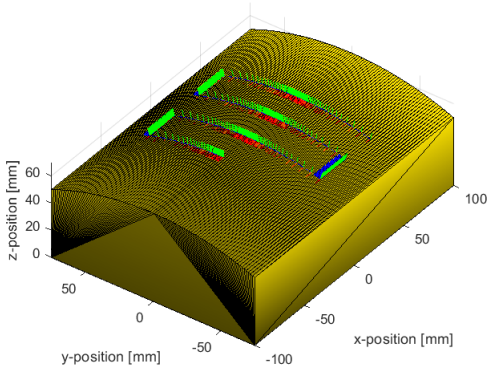


Figure 9.2: Planned trajectory on the phantom. Poses are defined by frames with red arrows representing the x-axis, blue arrows representing the y-axis and green arrows representing the negative z-axis.

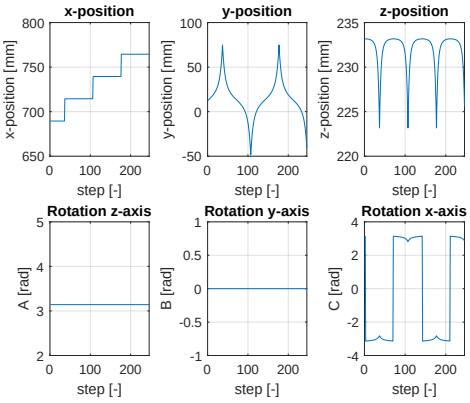


Figure 9.3: Pose information of the zigzag trajectory. Euler angles A, B and C are defined corresponding to Z-Y-X convention.

9.3 Experiment 2: Phantom for Ultrasound Scanning

During this project we also worked on the design and production of a phantom for ultrasound scanning. This experiment will provide a qualitative analysis of the quality of ultrasound images of the phantom. The expertise of a radiologist will be used to analyse the ultrasound images of the phantom. Also, we will compare the ultrasound images of the phantom to ultrasound images of a commercial phantom of Blue Phantom ™(figure 9.4), which is the world leader in ultrasound training simulation manikins for medical education. The result of this analysis will decide whether we will use our own phantom or the phantom of the Blue Phantom ™for the validation of the ultrasound-based visual servoing.

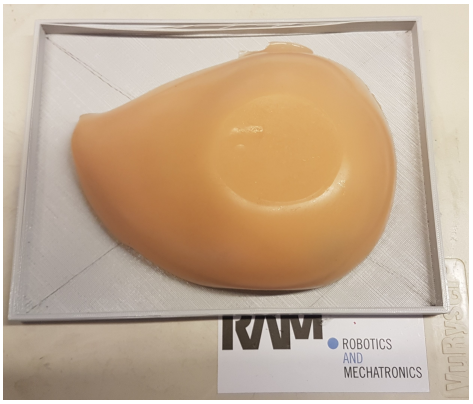


Figure 9.4: Commercial breast phantom of Blue Phantom ™.

9.4 Experiment 3: Ultrasound-Based Visual Servoing

The second milestone that we defined in chapter 1.3 to validate the proof of concept is the design and implementation of an ultrasound-based visual servoing algorithm. It was decided in chapter 6 to use confidence maps as visual servoing altorithm.

In the first part of this experiment we will examine the parameter settings and the control actions that result from the confidence maps. First we will acquire an ultrasound image of the phantom and examine the effect of the open parameters α, β and γ by changing the value of one parameter, while keeping the other two parameters constant. With this experiment, we obtain knowledge about the effect of the parameters on the appearance of the confidence maps. With this knowledge, we can optimize the parameter settings for ultrasound scanning of our phantom. Second, we will examine both control actions that result from the confidence

map of a single ultrasound image and investigate the relation between translation of the probe normal to the skin surface and the vertical displacement of the confidence-weighted barycenter. With this experiment, we can validate our visual servoing algorithm for probe manipulation.

In the second part of this experiment, we will investigate the control actions of our visual servoing algorithm while being executed by the SAIP controller. First, we increase the contact pressure by translation of the probe normal to the surface and observe the result on the confidence maps. From this, we will try to derive a relationship between the translation of the probe and the vertical position of the confidence-weighted barycenter. Second, we will investigate the control action for in-plane rotation as follows:

- Choose a robot configuration such that the probe makes contact with the phantom with only one side of its scanning surface.
- Acquire an ultrasound image and the perform confidence estimation on it.
- Report visual error and calculate the resulting control action for in-plane rotation.
- Let the robot execute the control action by using the SAIP controller.
- Acquire a new ultrasound image and perform the confidence estimation on it.
- Check whether the visual error has decreased.

10 Simulation

In this chapter we will describe the simulation model of the KUKA LBR Med robot manipulator. The simulation model is used to validate and test the SAIP controller before implementing it on the physical robot. This allows a faster design of the controller and reduces the risk of damage on the physical robot. In chapter 10.1 we will describe the content of the simulation model in more detail. In chapter 10.2 we will demonstrate the results of the simulation experiment to validate the controller design.

10.1 KUKA LBR Med Simulation Model

The model consists of three main parts: a trajectory generator, a controller and a model of the physical robot. The top level of the model is illustrated in figure 10.1. The trajectory generator generates a trajectory of desired end effector poses H^0_{ref}, which is passed on to the controller. The controller has, as described in chapter 7.2, the joint positions and joint velocities as feedback signals from the physical robot. The controller calculates the desired actuator torques that will be commanded to the physical robot, based on the difference between the reference end effector pose and the current end effector pose. The green box in figure 10.1 contains a list of global variables, like e.g. link lengths, link masses and initial stiffness and damping of the controller.

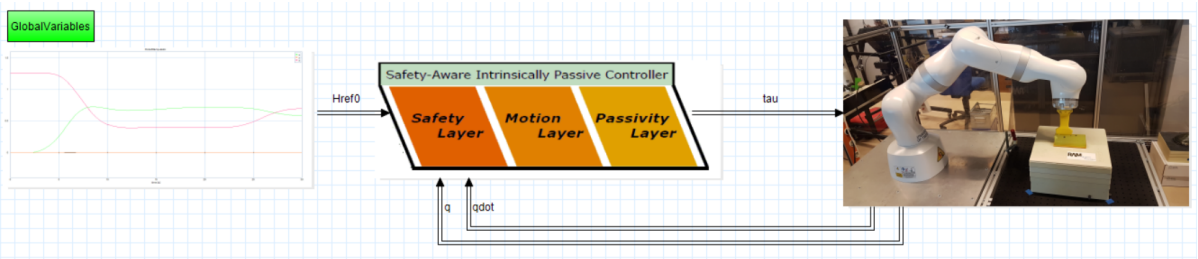


Figure 10.1: Top level of the 20-sim model of the KUKA LBR Med. The left block represents the trajectory generator, the middle block the controller and the right block the model of the physical robot.

10.1.1 Trajectory Generator

The controller design will be validated on a predefined trajectory that is characteristic for an ultrasound scanning task. The reference trajectory is divided in three phases: *StartPosition*, *Scanning* and *EndPosition*. The trajectory generator is a graphical model that consists of three parts: motion profiles, a joint trajectory generator and the direct kinematics (figure 10.2). The *StartPosition*, *Scanning* and *EndPosition* blocks contain time-dependent motion profiles to describe the motion of every joint for respectively the motion from the home pose to the start pose of the scanning trajectory, the scanning trajectory itself, and the motion from the end pose of the scanning trajectory to the end pose. Details about these reference trajectories will be discussed in more detail in chapter 10.2. The reference trajectories output a 7-dimensional time-dependent motion profile in degrees, who are merged by a mux element. The *JointTrajectory* block is an equation submodel that converts the joint trajectory to radians

and takes also the initial position of the manipulator into account. Finally, the *DirectKinematics* block is an equation submodel that takes the desired joint positions as input and calculates from this the desired end effector position H^0_{ref} by using equations (3.1) to (3.4).

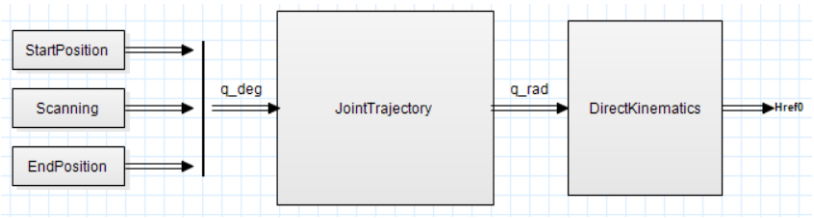


Figure 10.2: Building blocks of the trajectory generator.

10.1.2 Controller

The controller consists of two equation submodels: one that models the kinematics of the physical robot and one that implements the SAIP controller. This is illustrated in figure 10.3. The *KinematicsDynamics* block takes the current joint positions q, which are fed back from the physical robot, as input. From this, it calculates the current end effector pose H^0_{ee} by using equations (3.1) to (3.4) , the current geometrical Jacobian $J(q)$ by using equations (3.16) and (3.17), and the current mass matrix $M(q)$ by using equations (4.19) to (4.21).

The *SAIPController* block takes the current joint velocities $\dot{q}$ and the reference pose of the end effector as input. From this, the controller implements equations (4.21) to (4.31) such that the total energy of the system and the power that is transferred from the controller to the manipulator are limited according to the values of E_{max} and P_{max}. The updated stiffness matrices, co-stiffness matrices and damping matrix are used in equation (4.7) to (4.13) to calculate the desired controller torques for every joint. Finally, equations (5.11) to (5.14) are implemented to include the energy tanks which determine the controller torques that are commanded to the physical robot.

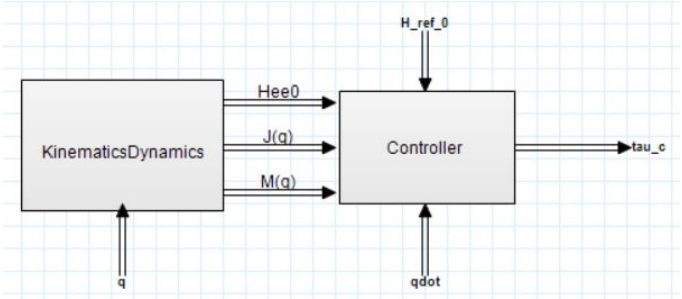


Figure 10.3: Building blocks of the controller.

10.1.3 Model of the Physical Robot

The model of the KUKA LBR Med is built according to the port-Hamiltonian model of chapter 7. The KUKA LBR Med is an articulated robot with 7-DOFs, so the same amount of joint blocks and link blocks are used (figure 10.4). The 1^{st}, 3^{rd}, 5^{th} and 7^{th} joint rotate around the z-axis, according to coordinate frame Ψ_0 of figure 8.2. The 2^{nd} and 6^{th} joint rotate around the positive y-axis and the 4^{th} joint rotates around the negative y-axis. An external wrench is implemented by a torque and a force on the end effector according to equation (3.18).

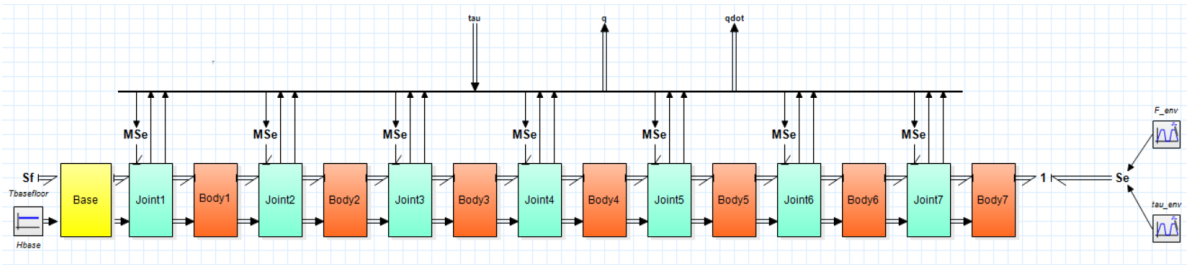


Figure 10.4: Building blocks of the port-Hamiltonian simulation model of the KUKA LBR Med.

10.2 Simulation Results

In this simulation experiment we first perform an experiment in which we check whether the joint limitations are correctly implemented in the robot model. Subsequently, we will validate the controller by an experiment that mimics a scanning task.

10.2.1 Joint Limits Experiment

In chapter 7.2 we described the implementation of joint limits in the manipulator model by a virtual wall. In this experiment we start with the robot in the configuration at the end of the *StartPosition* phase: $q = (0; 45; 0; -60; 0; 75; 90)^T$. Subsequently, the 4^{th} joint is commanded to -180 degrees, while its joint limit is on -120 degrees. The resulting motion of the 4^{th} joint is given in figure 10.5. The joint position of the 4^{th} joint is limited to -120 degrees, while the commanded position is further decreased. It must be noted that this limitation results from the joint limitation in the port-Hamiltonian robot model, and is not necessarily implemented by the controller. However, it is necessary for validation of the controller that the joint limitations are also considered in simulation.

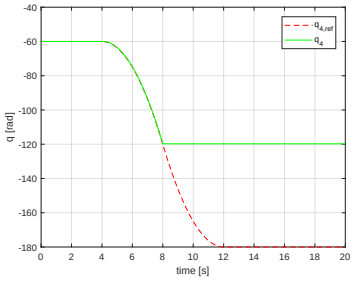


Figure 10.5: Joint limitation of the 4^{th} joint of the robotic manipulator for a commanded joint motion from -60 degrees to -180 degrees.

10.2.2 Scanning trajectory experiment

In this experiment we use the trajectory that was described in chapter 9.1. The SAIP controller is used to control the robot, with the parameters that are reported in table 10.1. The reference trajectory and actual trajectory in joint space are given in figure 10.6. The blue vertical lines indicate transitions between the phases, so every phase lasts 10 seconds. Only the joints with non-zero reference trajectory are reported. The Cartesian error, which is defined as the Euclidean distance between the end effector and the reference position, is given in figure 10.7.

Table 10.1: Controller parameters that are used in the simulation.

Symbol	Description	Value	Unit
E_{max}	Energy limit	0.2	J
P_{max}	Power limit	0.1	W
ϵ	Energy threshold in tank	0.001	J
s	Initial tank state	1	J
K_o	Translational stiffness	300	N/m
K_t	Rotational stiffness	5000	Nm/rad
K_c	Coupling stiffness	200	N/rad
b	Damping coefficient	5	-

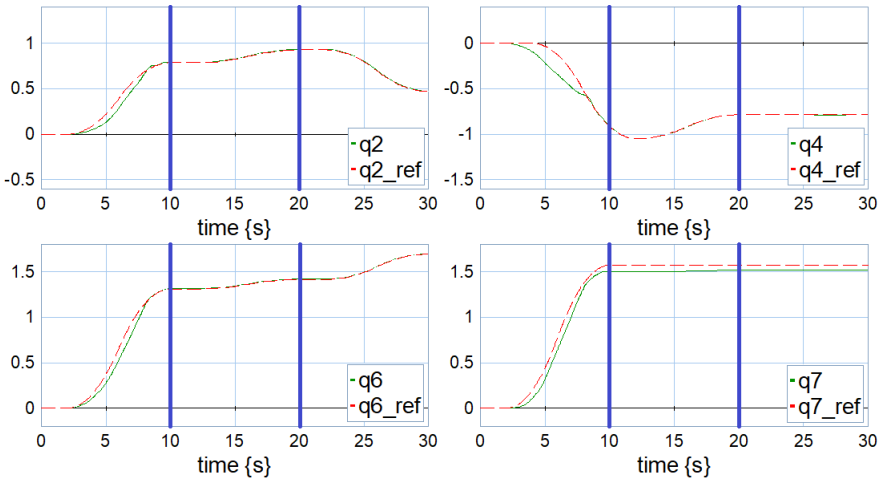


Figure 10.6: Reference and actual joint positions in the simulation for joint 2, joint 4, joint 6 and joint 7. Joint positions are given in radians.

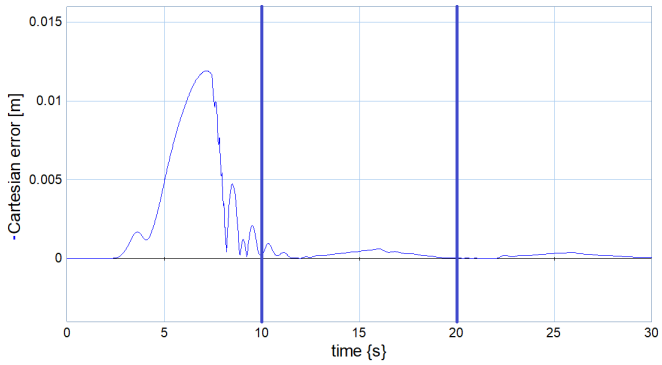


Figure 10.7: Cartesian error between the end end effector and the reference position in the simulation experiment.

The Cartesian error is the largest in the *StartPosition* phase, with a maximum of about 12 mm. This is due to the fact that safety limits are reached, such that the controller adjusts its behavior in this period. In figure 10.8 is reported how the energy limit is reached during the *StartPosition* phase. The value of λ is adjusted to scale the spring stiffnesses to limit the actual energy of the system to a value of 0.2 J. A little peak in energy is visible in the *EndPosition* phase between 24 and 28 seconds. However, this peak does not exceed the limit of 0.2 J, so the value of λ is not adjusted there.

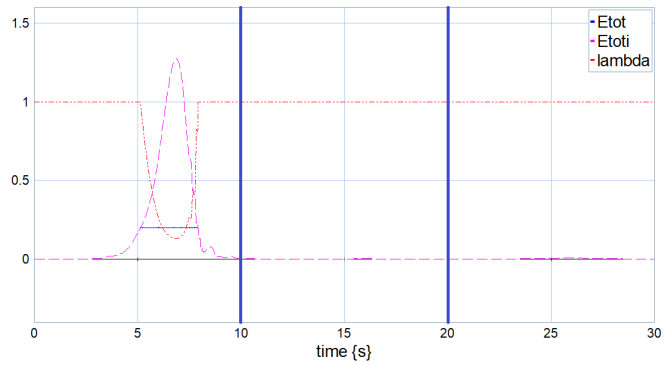


Figure 10.8: Limitation of the total energy of the robot by the SAIP controller. The actual total energy E_{tot} and the initial total energy $E_{tot,i}$ are given in J and scaling factor λ is dimensionless.

In figure 10.9 is illustrated how the power limit is reached during the *StartPosition* phase. The value of β is adjusted to limit the actual power that is flowing from the controller to the robotic manipulator to a value of 0.1 W.

Finally, the controller torques are reported in figure 10.10. We only reported the torques of the joints with non-zero reference position. It follows that the controller torques are relatively high in the *StartPosition* phase compared to the rest of the motion, which coincides with the range of this motion. The joint torques are oscillating for joint 2, joint 4 and joint 6, but this results in a smooth motion in joint space as illustrated in figure 10.6.

The simulation results show that the controller is working correctly and that energy limitations are considered. In chapter 11 we will implement the controller on the physical robot and perform experiments with it.

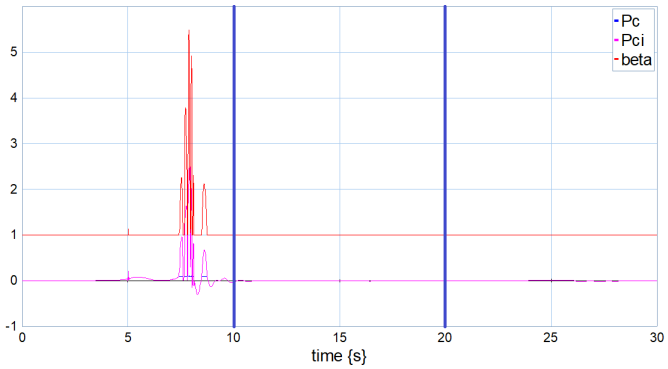


Figure 10.9: Limitation of the power flowing from the controller to the robotic manipulator by the SAIP controller. The actual power P_c and the initial power $P_{c,i}$ are reported in W and scaling factor β is dimensionless.

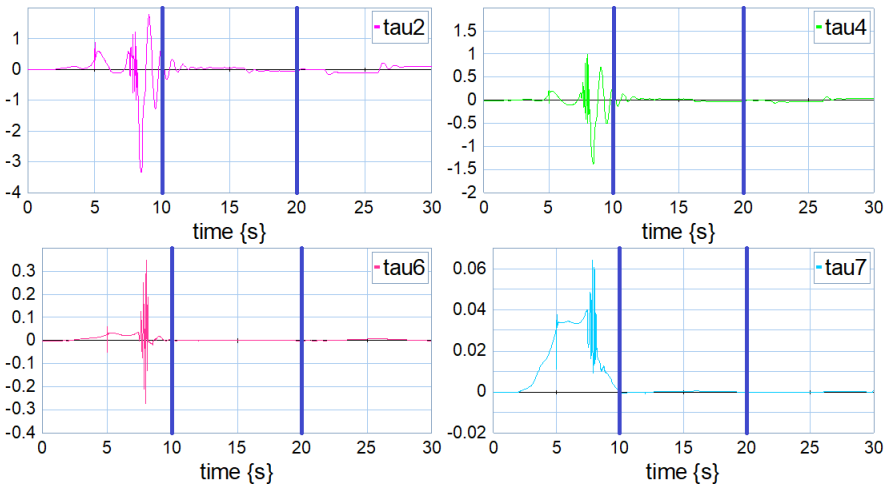


Figure 10.10: Controller torques of joint 2, joint 4, joint 6 and joint 7 in Nm.

11 Results and Discussion

This chapter provides an experimental validation to proof the concept of our proposed control strategy for autonomous ultrasound imaging. In chapter 1.3 we defined two main tasks, being: "Design and implementation of a compliant control strategy" and "Design and implementation of an ultrasound-based visual servoing". Both will be examined through experiments with the KUKA LBR Med. An additional experiment will be dedicated to the examination of the design and production of our phantom for ultrasound scanning. The results will determine whether the phantom that we created is suited for the experiments that involve ultrasound imaging.

11.1 Experiment 1: Compliant Control Strategy

This section provides an experimental validation of the design and implementation of the SAIP controller, where we will focus on the dynamic behaviour of the robot and its safety limitations. We will evaluate the controller on its behaviour in four experiments: setpoint experiment, disturbance experiment, energy tank experiment and a scanning trajectory experiment. The results of these experiments will be discussed in the remaining of this section and provide an insight in the features of the SAIP controller. The visual servoing algorithm is not yet considered.

11.1.1 Setpoint Experiment

This experiment shows the behaviour of the robot for a commanded translation and a commanded rotation, without interaction with the environment. These motions are equal to the control actions that result from the confidence maps, as described in chapter 6.3. The robot starts in the configuration $q = (-0.08, 0.53, 0.22, -0.70, -0.14, 1.35, 1.62)^T$ rad, with $q \in \mathbb{R}^7$ a vector of joint positions. After 1.5 seconds the setpoint is changed stepwise to a point that is 0.08 m lower in the global z-direction (see figure 8.2). After 8 seconds a rotation of -0.2 rad around the y-axis of the probe, which is an in-plane rotation, is commanded to the robot.

In the experiments we used the initial set of controller parameters from table 11.1. To examine the influence of the different parameters on the behaviour of the controller, we changed the value of either E_{max}, P_{max}, K_o, K_t, K_c and b, while keeping the other parameters constant.

The effect of E_{max} on the behaviour of the robot is illustrated in figure 11.1. It follows that λ is in general lower for a lower value of E_{max}, because the maximum energy is exceeded earlier which results in a reduction of the stiffness values K_t, K_o and K_c. This increased compliance of the robot results in a higher translational and rotational error. However, this is not necessarily negative, since the robot has to behave compliant in the interaction with the environment. However, when E_{max} is too low the robot has not enough energy available to converge to the

Table 11.1: Initial controller parameters for the setpoint experiment.

E_{max}	P_{max}	ϵ	s	K_o	K_t	K_c	b
1.5 J	0.5 W	0.001 J	1.0 m	100 Nm/rad	2000 N/m	0 N/rad	10

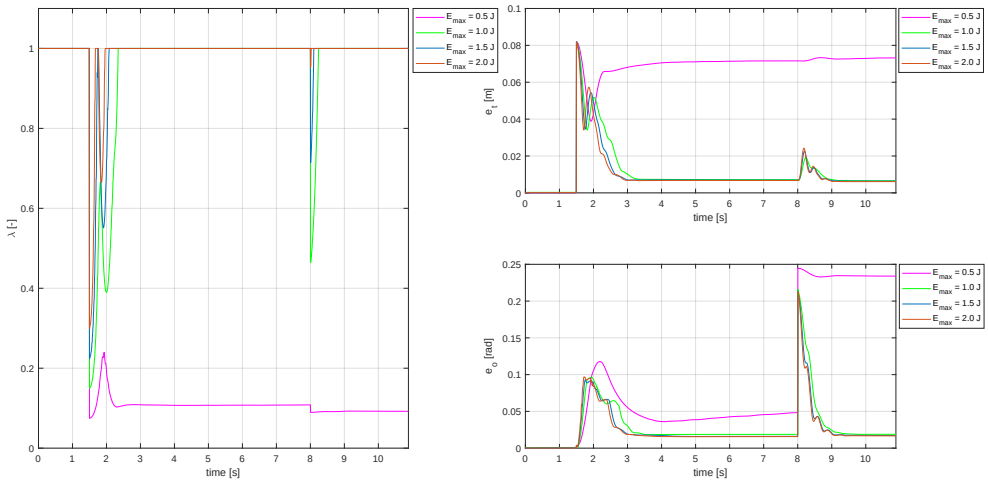


Figure 11.1: Scaling parameter λ, translational error e_t and orientational error e_o during the setpoint experiment for different values of E_{max}.

setpoint, as was the case for E_{max} = 0.5 J. Summarizing, E_{max} has to be chosen such that it provides sufficient energy to the robot to converge to a setpoint, but also gives the robot a desired compliance in case that the total energy of the robot becomes potentially unsafe for its environment.

The effect of P_{max} on the behaviour of the robot is illustrated in figure 11.2. When the setpoint is changed, the power flowing from the controller to the robot increases due to increasing joint torques and joint velocities. The upper figure shows how the controller scales β in order to increase the damping to limit the total power. When P_{max} is lower, more damping is added to the joints for an equal change in setpoint. The additional damping also results in a decrease of kinetic energy, due to lower joint velocities. The lower figure shows that the kinetic energy of the robot is in general lower for lower values of P_{max}.

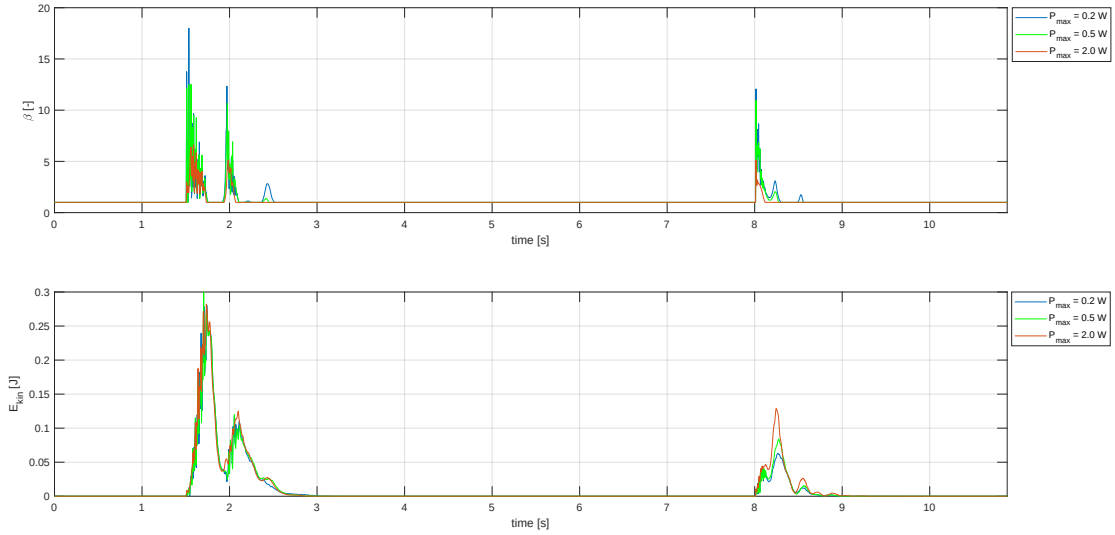


Figure 11.2: Scaling factor β (upper figure) and kinetic energy E_{kin} (lower figure) for different values of P_{max}.

The effect of the spring stiffnesses K_o, K_t and K_c is illustrated in figure 11.3. The 3×3 matrices K_o, K_t and K_c of equation (4.7) are chosen such that they are diagonal matrices with equal elements on the diagonal. The first row of figures shows that the orientational energy is very dependent on the value of K_o. Whenk K_o is too low, the spring is not strong enough to bring the end effector to the setpoint, which results in a large orientational error and a residual potential orientational energy. The second row of figures shows a similar behaviour, but now for the translation. When the value of K_t is high, the end effector starts oscillating around the setpoint. The third row of figures shows that a low value of K_c (or even a value of 0) reduces the orientational and translational error. It can be concluded that high values of K_o and K_t and a low value of K_c result in lower pose errors with respect to the predefined trajectory. As mentioned before, this is not necessarily beneficial for the interaction during ultrasound scanning, because the robot sometimes has to behave compliant to correct for movements of the patient. So a trade-off must be made between the importance of compliance and trajectory following.

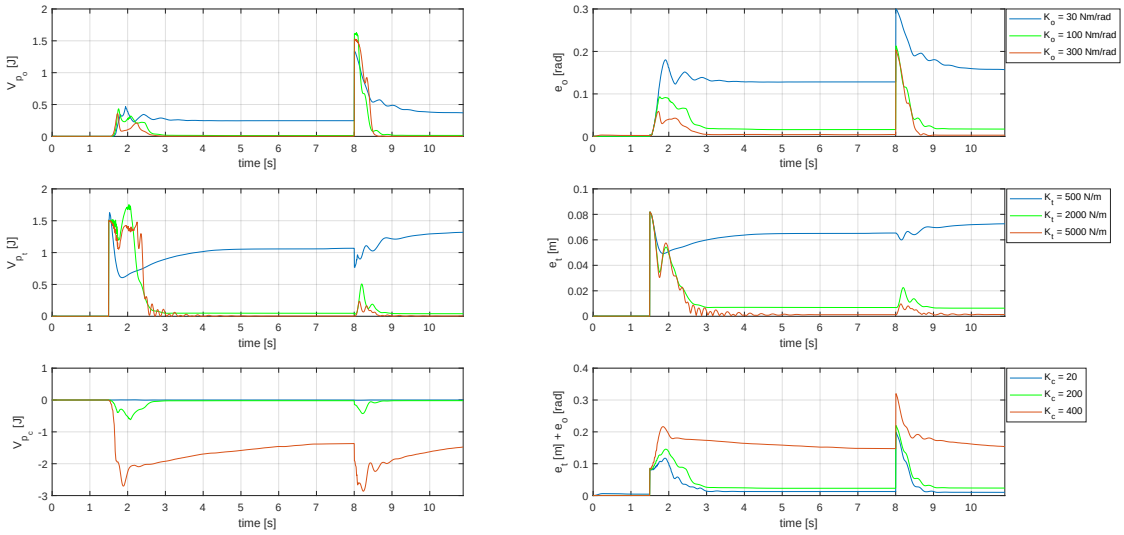


Figure 11.3: Effect of K_o, K_t and K_c on respectively the potential orientational energy V_{p_o} and orientational error e_o, potential translational energy V_{p_t} and translational error e_t and potential coupling energy V_{p_c} and translational plus orientational error $e_t + e_o$.

Finally, the effect of the inital damping parameter b is illustrated in figure 11.4. It strikes that a low value of b results in more oscillation, but converges to a smaller translational and orientational error. The bottom figure explains this phenomenon. For a low value of b, the controller automatically increases the scaling factor β, such that damping is only added when the power exceeds the maximum value P_{max}. This results in a faster control with lower residual pose error.

The experiments in this section have shown how the behaviour of the robot can be tuned by several parameters. The ideal settings depend on the task that has to be executed. For autonomous ultrasound scanning, there is a trade-off between the compliance of the robot and the importance of trajectory following. For the rest of the experiments, we will use the values of table 11.1.

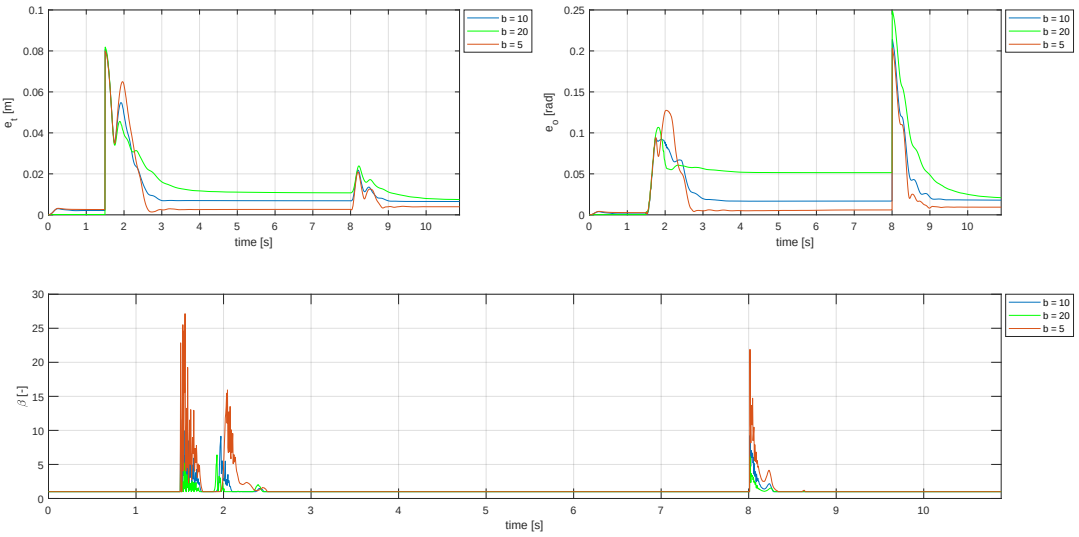


Figure 11.4: Effect of the initial damping parameter b on the translational error e_t (upper left), orientational error e_o (upper right) and scaling factor β (bottom).

Discussion

This experiment has shown that our control is capable of rotating and translating the ultrasound probe, while incorporating the specified safety limitations. This feature can be used by the visual servoing algorithm, that controls the in-plane rotation of the probe and the translation normal to the skin surface. It was also shown that the parameters can be tuned to yield different behaviour of the robot. In this way, the controller can be tuned to provide a safe human-robot interaction for different situations.

Disturbance Experiment

The disturbance experiment shows the behaviour of the robot under external disturbances. The setpoint of the robot corresponds to the initial configuration of the robot in the setpoint experiment and the parameter settings are equal to the ones in table 11.1, but with a coupling stiffness of K_c = 200 N/rad. After about 3 seconds the robot is pushed at its end effector in the global y-direction, after about 8 seconds a torque is applied to the end effector and after about 16 seconds both a translation and rotation is applied to the end effector of the robot. The results are given in figure 11.5.

The upper left figure shows how the translational and rotational error grow as a result of the external disturbances. The upper right figure shows how the scaling parameter λ decreases during the disturbances, which results in a more compliant behaviour of the robot.

From the bottom right figure it follows that the first disturbance results in a large increase in kinetic energy and translational potential energy, together exceeding the maximum energy E_{max}. This causes a potentially dangerous situation according to the value of E_{max} that is defined as 1.5 J. As a result, the controller decreases the value of λ and consequently decreases the stiffness values, in order to decrease the potential energy due to the spatial spring. In this way, the total energy of the system is limited to a value that is equal to E_{max}. Similar behaviour can be noticed at the second disturbance, although the rotational potential energy is now dominant. This makes sense, since a torque was applied to the end effector, where the first disturbance was more like a linear force. The third disturbance is a combination of both a torque and a force, which results in rotational, translational and coupling potential energy.

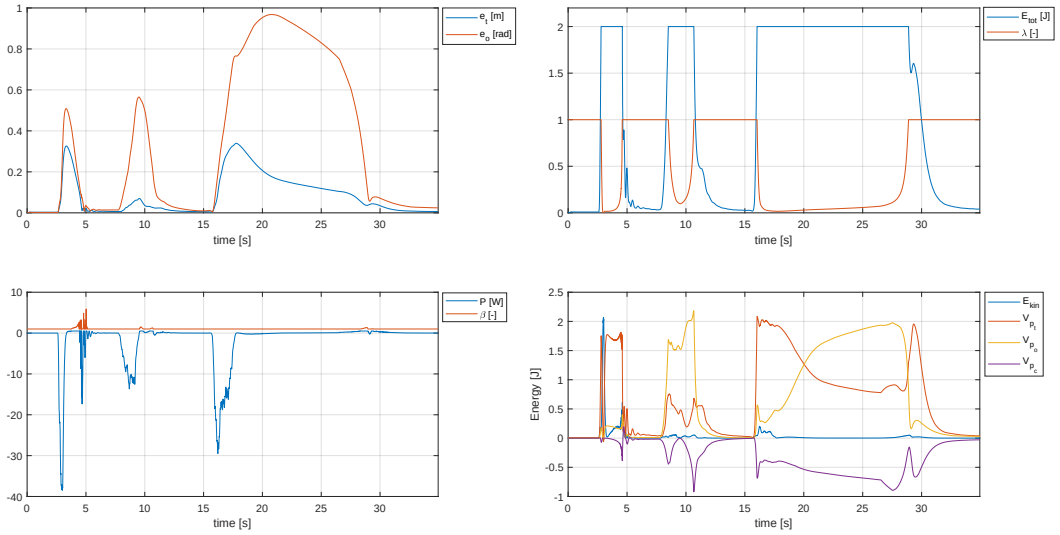


Figure 11.5: Translational error e_t and rotational error e_o (upper left), total energy E_{tot} and scaling factor λ (upper right), power P and scaling factor β (lower left) and kinetic energy E_{kin} and potential energies $V_{p_t}, V_{p_o}, V_{p_c}$ (lower right) during the disturbance experiment. A linear force is applied to the end effector after about 3 seconds, a torque is applied after about 8 seconds and both a translation and rotation are applied after about 16 seconds.

Finally, the bottom left figure shows that power is transferred from the environment to the robot when the disturbances are applied to the end effector. When the robot returns to the setpoint after the push disturbance, it contains kinetic energy that flows back from the robot to the controller. This power exceeds the maximum value of P_{max} so the controller adjusts β in order to decrease the angular velocity of the joints. In this way, the power is limited such that it avoids potentially dangerous situations and it also decreases the amount of oscillation around the setpoint.

Discussion

This experiment has shown that the SAIP controller allows the robot to behave compliant under external disturbances by monitoring the total energy and power of the system. The controller automatically adjusts the scaling factors to achieve desired behaviour in different situations. In this way, the robot is able to follow a scanning trajectory with appropriate compliance and increases its compliance when the patient moves or something in the environment disturbs the motion of the robot.

11.1.2 Energy Tanks Experiment

The goal of this experiment is to demonstrate how the energy tanks get depleted during a task and how the robot behaves when a tank is empty. The setpoint during this experiment describes a block wave on the first joint with an amplitude of 0.6 rad and a period of 2.5 seconds, starting at $q = (0.30, 0.53, 0.22, -0.70, -0.14, 1.35, 1.62)^T$ rad. This results in a circular motion of the end effector in the horizontal global plane. The results of this experiment are illustrated in figure 11.6.

The upper left figure shows the depletion of the energy tank of the first joint. Every time the setpoint changes, the first joint uses a part of its available energy. The red line represents ϵ, i.e. the minimum required energy in the tank. The upper right figure shows the difference between the actual position of joint 1 and its reference position. When the tank of the first joint is empty

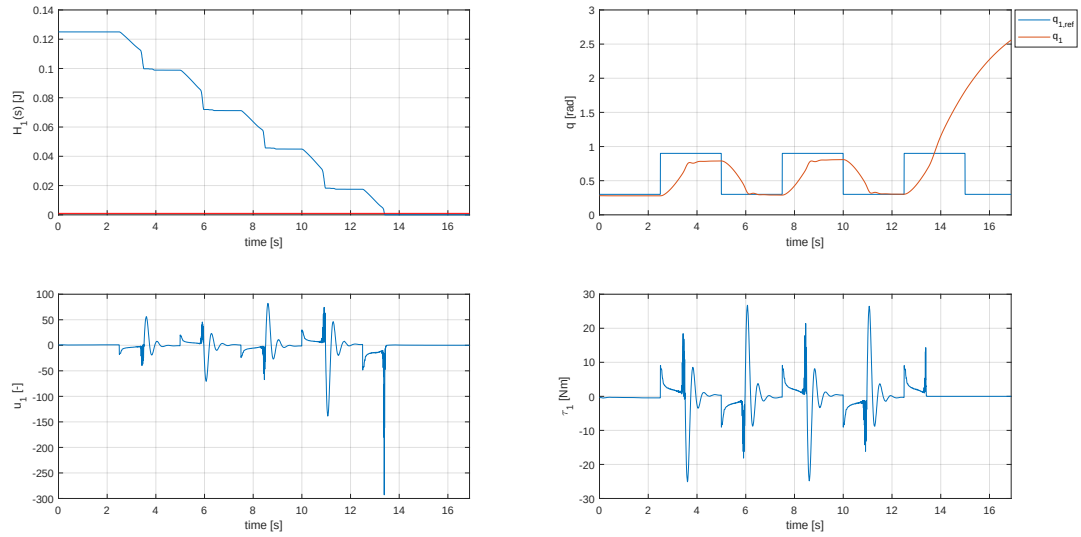


Figure 11.6: Results of the energy tanks experiment. Upper left: depletion of the energy tank of the first joint, upper right: reference position and actual position of the first joint, lower left: transmission ratio of the first joint, lower right: joint torque of the first joint.

(after about 13 seconds), there is no energy left in the tank to execute a task so the first joint deviates from its reference position due to its current kinetic energy. The bottom left figure shows how the transmission ratio is adjusted according to equation (5.14). The bottom right figure shows the joint torque of the first joint, which becomes zero after the tank of the first joint is depleted.

Discussion

This experiment validates the concept of passivity. It has been demonstrated that a joint can not use more energy than it has available in its energy tank. When the energy tank is empty, the joint is virtually shut down and can be moved freely. This effectively prevents the robot from becoming unstable.

Imagine a case where there is a bug in the reference pose of the end effector of the robot, such that there is a large Euclidean distance between two consecutive poses. When the safety limitations are high, the total energy of the robot could become dangerous for its environment. However, when the energy level of the energy tanks is properly monitored by the Supervisor, they will only provide a certain amount of energy to the joints such that the robot will never get to a state that is potentially dangerous for its environment. This is an important feature in the interaction between a robot and a patient.

11.1.3 Scanning Trajectory Experiment

The goal of this experiment is to demonstrate the behaviour of the robot and the controller for trajectory following. We will use the block-shaped trajectory that is described in chapter 9. This trajectory is executed on the thigh phantom that we created with a reference orientation that is normal to the surface of the phantom.

First, we have executed the scanning trajectory with the impedance controller from Sunrise Workbench for three different parameter settings. It must be noted that the damping b is limited to a relative value between 0 and 1 in Sunrise. The impedance controller is equivalent to the one described in chapter 4.2.4. The results are given in figure 11.7. The mean translational errors e_t for the different parameter settings are (from the top down): 3.0 mm, 8.5 mm and 26.3

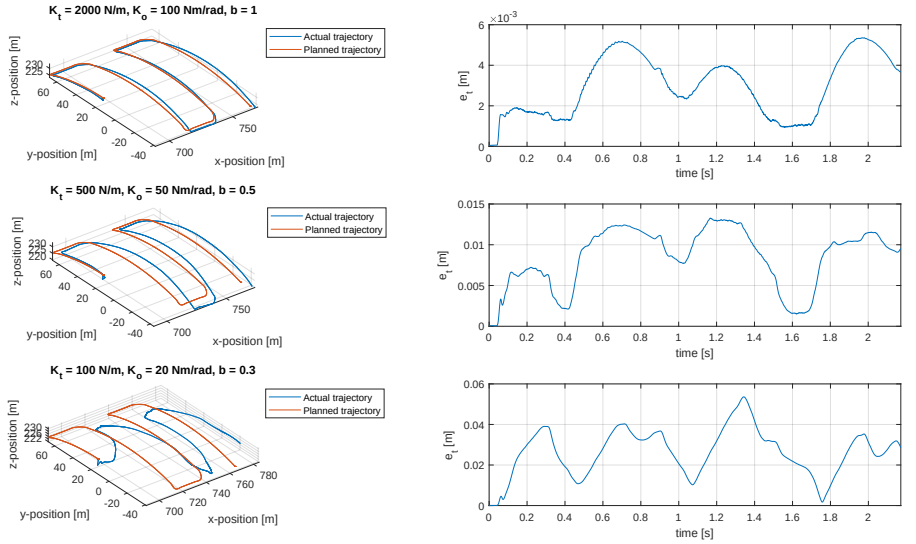


Figure 11.7: Actual and planned trajectory and translational and orientational error for three different parameter setting of the impedance controller from Sunrise Workbench. The mean translational errors e_t for the different parameter settings are (from the top down): 3.0 mm, 8.5 mm and 26.3 mm.

mm. As expected, the translational error increases when the stiffness and damping decrease. Unfortunately, no information about the orientation of the end effector can be retrieved from Sunrise.

It is now interesting to compare these results with trajectory following with the SAIP controller. Similar parameter settings are used for the first two trajectories. The damping b is estimated to yield similar behaviour compared to the scaled damping in Sunrise. The results are given in figure 11.8. The mean pose errors for the different parameter settings are (from the top down): ($e_t = 4.1$ mm, $e_o = 0.0140$ rad), ($e_t = 15.2$ mm, $e_o = 0.0229$ rad) and ($e_t = 4.3$ mm, $e_o = 0..0070$ rad). It follows that the translational error is only slightly higher with the SAIP controller for similar parameter settings as the impedance controller from Sunrise. By increasing the values of the parameters it is possible to decrease also the orientational error, but this does not necessarily decrease the translational error any further. From figure 11.7 and 11.8 it can be concluded that the SAIP controller is able to perform similar as the impedance controller from Sunrise on trajectory following. The big advantage of using the SAIP controller is that it also takes care of safety limitations and that it ensures passivity of the overall system.

The behaviour of the SAIP controller is monitored for the last set of parameter settings (K_t = 5000 N/m, $K_o = 300$ Nm/rad, $b = 10$). The total energy was limited to $E_{max} = 0.2$ J and the power was limited to $P_{max} = 0.5$ W. The results are given in figure 11.9. The safety limitations were not exceeded during the scanning procedure, which is acceptable since there were no external disturbances. The kinetic and potential energy fluctuates depending on the position on the trajectory. The potential coupling energy was always zero, since $K_c = 0$ N/rad was used. The joint torques stayed bounded during the whole scanning trajectory. The joint torques were highest for the first and second joint, which makes sense since they have to carry the mass of the distal links.

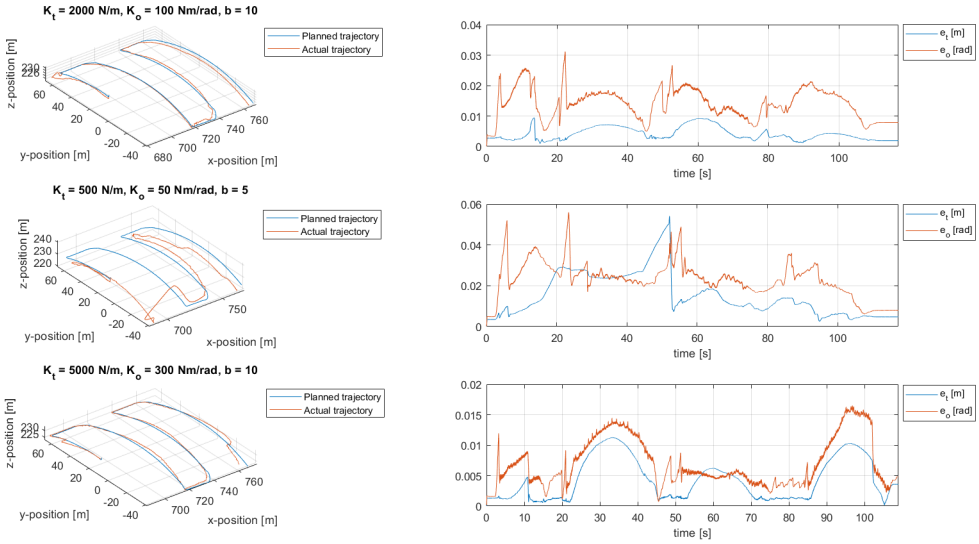


Figure 11.8: Actual and planned trajectory and translational error for three different parameter setting of the SAIP controller. The mean pose errors for the different parameter settings are (from the top down): (e_t = 4.1 mm, e_o = 0.0140 rad), (e_t = 15.2 mm, e_o = 0.0229 rad) and (e_t = 4.3 mm, e_o = 0..0070 rad).

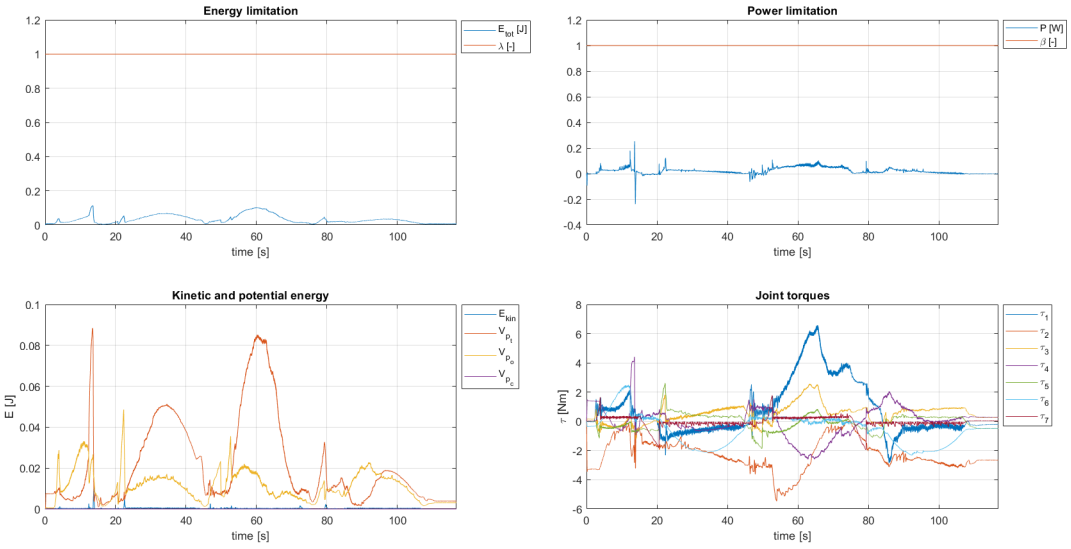


Figure 11.9: Behaviour of the SAIP controller during scanning of the thigh phantom. Upper left: total energy of the robot and scaling factor λ, upper right: power flowing from the controller to the robot and scaling factor β, lower left: kinetic and potential energy, lower right: joint torques.

Discussion

This experiment validates that the SAIP controller yields sufficient performance on trajectory following, while it implements safety limitations and guarantees passivity in case of interactions with the environment. This makes the control suitable for autonomous ultrasound scanning of the breast. In experiment 3 we will add the visual servoing algorithm in combination with the SAIP controller to improve the image quality online.

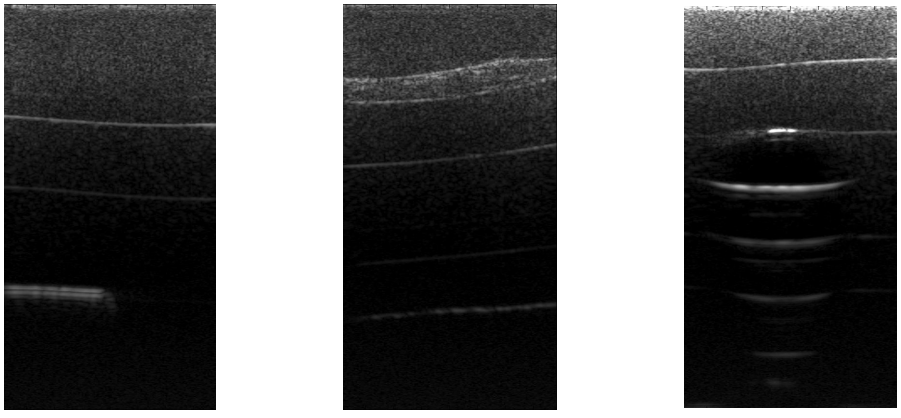
11.2 Experiment 2: Phantom for Ultrasound Scanning

This experiment serves as an evaluation of the ultrasound phantom that we produced. Details about the design and production of the ultrasound phantom were described in chapter 8.3. We will first discuss the structure and texture of the phantom and, second, we will compare ultrasound images of our phantom with the commercial phantom of Blue Phantom ™.

The surface of the phantom is more sticky than a human skin, which can introduce additional friction between the probe and the phantom compared to scanning of human body parts. However, applying ultrasound gel reduces the friction resulting in a realistic amount of friction compared to ultrasound scanning on the human skin. The structure of the phantom resembles the human thigh, due to the softer layer that mimics fat just beyond the skin. This gives the phantom an appropriate stiffness and, at the same time, rebound properties that are very similar to the human thigh.

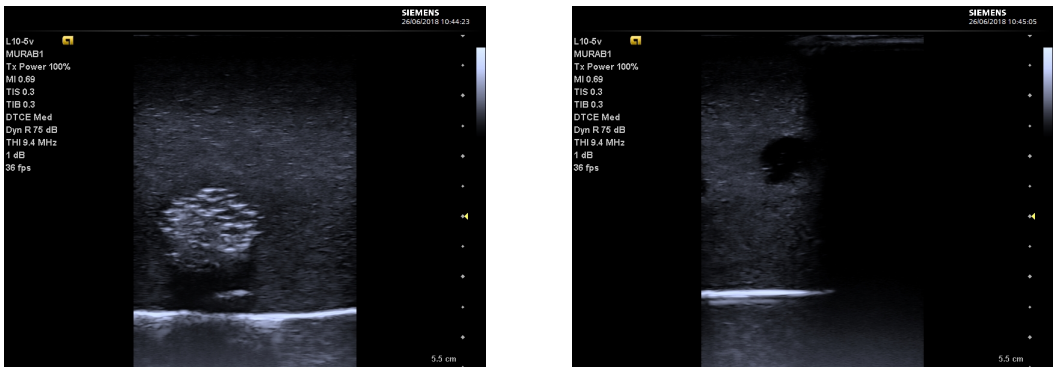
Second, we captured ultrasound images of the thigh phantom and discussed them with a radiologist. Some images are illustrated in figure 11.10. We can recognize some of the elements from figure 8.11, being the bone in figure 11.10a, an inlay in figure 11.10b and the fish oil capsule in figure 11.10c. However, there is a lot of attenuation in the images, which has two major disadvantages. First, the detail resolution and contrast resolution are in general very low at the bottom of the image. Second, it requires a lot of contact force between the probe and the skin to visualize the different elements. The latter collides with the requirements for safe human-robot interaction.

Another set of images was captured from the Blue Phantom ™, which are given in figure 11.11. Figure 11.11a is captured by just placing the probe on the phantom without applying additional contact force. This already yields a clear image that is suited for further examination. Figure 11.11b is captured with only one side of the contact surface of the ultrasound probe in contact with the phantom. This is a typical situation where the visual servoing algorithm should correct for the in-plane rotation. Figure 11.11b clearly visualizes which part of the ultrasound probe is in contact with the phantom, which makes it suitable for our visual servoing algorithm.



(a) **(b)** **(c)**

Figure 11.10: Ultrasound images of the thigh phantom: **(a)** Different layers and the bone segment **(b)** Different layers and an inlay at the top **(c)** Fish oil capsule with reflection phenomena.



(a) **(b)**

Figure 11.11: Ultrasound images of the Blue Phantom ™: **(a)** Image captured with minimal contact force between the phantom and the probe **(b)** Image captured with only half of the contact surface of the probe in contact with the phantom.

Discussion

Based on the results in this section we decided to use the Blue Phantom ™ for our experiments that involve ultrasound imaging. It provides clear images with minimal contact force which are very suitable for our visual servoing algorithm. Nevertheless, the production process of the thigh phantom provides more flexibility in the design of the phantom and can be potentially useful if the attenuation of the ultrasound waves inside the phantom is reduced.

11.3 Experiment 3: Ultrasound-Based Visual Servoing

This experiment provides an experimental validation of the design and implementation of our ultrasound-based visual servoing algorithm. It was proposed in chapter 6.1 to use confidence maps for visual servoing. We will first evaluate confidence maps and ,secondly, we will evaluate the visual servoing algorithm by robotically manipulating an ultrasound transducer according to the confidence maps.

11.3.1 Confidence Maps

In this section we will discuss the effect of the open parameter settings of the confidence maps and the control actions that result from the confidence estimation. This control action will be used as input of the controller to manipulate the probe.

Parameter Settings

The confidence estimation problem has three open parameters, being α, β and γ. We will examine the effect of these parameters on the confidence maps of an ultrasound image that includes tissues of different echogenicity. Figure 11.12 shows an ultrasound image of the phantom and the corresponding confidence map with initial parameter settings. For now, the red line, green cross and numbers can be ignored. We have chosen $\alpha = 2$, $\beta = 90$, $\gamma = 0.2$ as initial parameter settings, because it provides reasonable contrast in confidence in both the vertical and horizontal direction for this image.

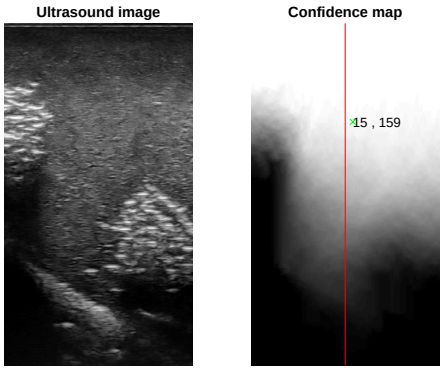


Figure 11.12: Ultrasound image of the phantom and corresponding confidence map for the initial parameter settings $\alpha = 2$, $\beta = 90$, $\gamma = 0.2$.

The parameter α controls the importance of acoustic attenuation, so it effectively scales the confidence in the vertical direction. Figure 11.13 shows the effect of α on the confidence map. It follows that the confidence decays faster in vertical direction for a higher value of α.

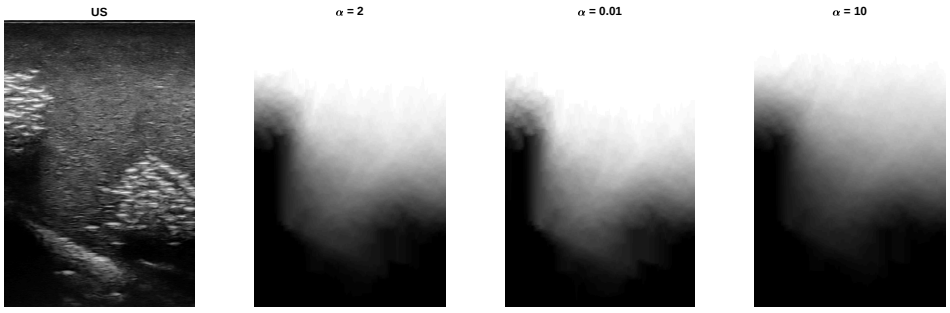


Figure 11.13: Effect of the α parameter on the confidence map with fixed values of $\beta = 90$ and $\gamma = 0.2$.

The parameter β controls the importance of acoustic absorption, which effectively scales the contrast in the confidence map. Figure 11.14 shows the effect of β on the confidence map. It follows that the contrast increases for lower values of β and decreases for higher values of β.

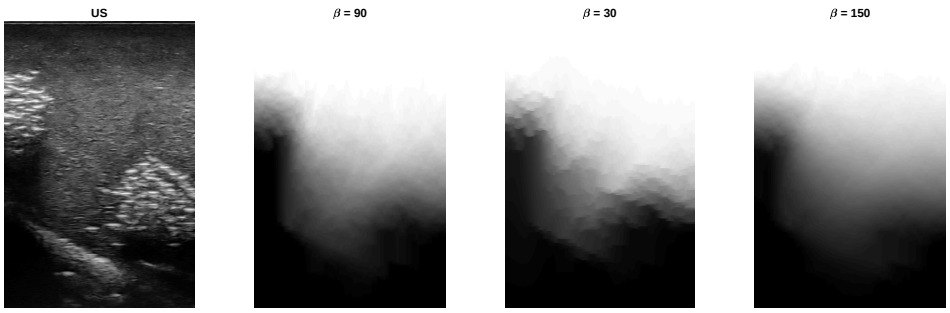


Figure 11.14: Effect of the β parameter on the confidence map with fixed values of $\alpha = 2$ and $\gamma = 0.2$.

The parameter γ penalizes horizontal and diagonal random walks. Tuning this parameter requires a trade-off between having distinct confidence and having minimal discontinuities in horizontal direction. Figure 11.15 shows the effect of γ on the confidence map. It follows

that the discontinuity in horizontal direction increases for higher values of γ, while it becomes more difficult to distinguish between areas with different confidence for lower values of γ.

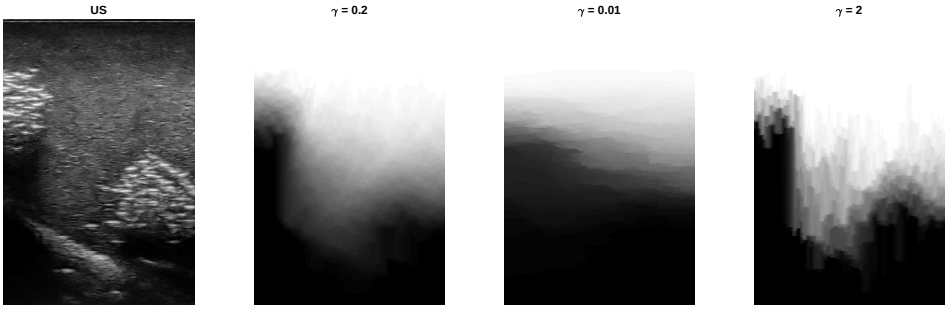


Figure 11.15: Effect of the γ parameter on the confidence map with fixed values of $\alpha = 2$ and $\beta = 90$.

Discussion

The optimal parameter settings are dependent on the internal structure of the scanned volume. By examining the effect of different parameter settings, we have obtained knowledge about their influence on the appearance of the confidence maps. With this knowledge, we can optimize the parameter settings for ultrasound scanning of our phantom. For the Blue Phantom ™ we will use $\alpha = 2$, $\beta = 90$ and $\gamma = 0.2$.

11.3.2 Control Actions

The confidence maps will be used to control the in-plane rotation of the probe and the translation normal to the skin. The red line in figure 11.12 represents the central scan line and the green cross represents the confidence-weighted barycenter of the image. The coordinates indicate the horizontal pixel position wit respect to the central scan line and the vertical pixel position of the confidence-weighted barycenter. These coordinates will be used to calculate the control actions with the equations of chapter 6.3.

The experiment is divided in two phases. First, we investigate the control action for translation normal to the skin and investigate its relation with the vertical displacement of the confidence weighted barycenter. Second, we investigate the control action for in-plane rotation of the probe.

Translation Normal to Skin Surface

We will investigate whether the control action for translation normal to the skin surface can be used for vertical positioning of the confidence-weighted barycenter (e.g. for localization of a target). In this experiment, the pose of the probe was fixed except for its position in global z-direction (normal on the surface of the phantom). In this way, the contact pressure can be increased by decreasing the position of the probe in z-direction. We have captured images while stepwise increasing the z-position of the probe to investigate the effect of the translation of the probe normal to the skin surfacce on the vertical position of the confidence-weighted barycenter. The ultrasund probe was at all times in contact with the phantom. The resulting ultrasound images and corresponding confidence maps are given in figure 11.16. A possible target object is positioned in the bottom left of the figure.

It follows that the vertical position of the confidence-weighted barycenter decreases (moves towards the top of the image) for increasing contact pressure. Intuitively, one could expect that the confidence-weighted barycenter would move towards the bottom of the image due to the fact that an increasing contact pressure enhances the confidence in the lower region of the image. However, by increasing the contact pressure, the underlying tissue is compressed which

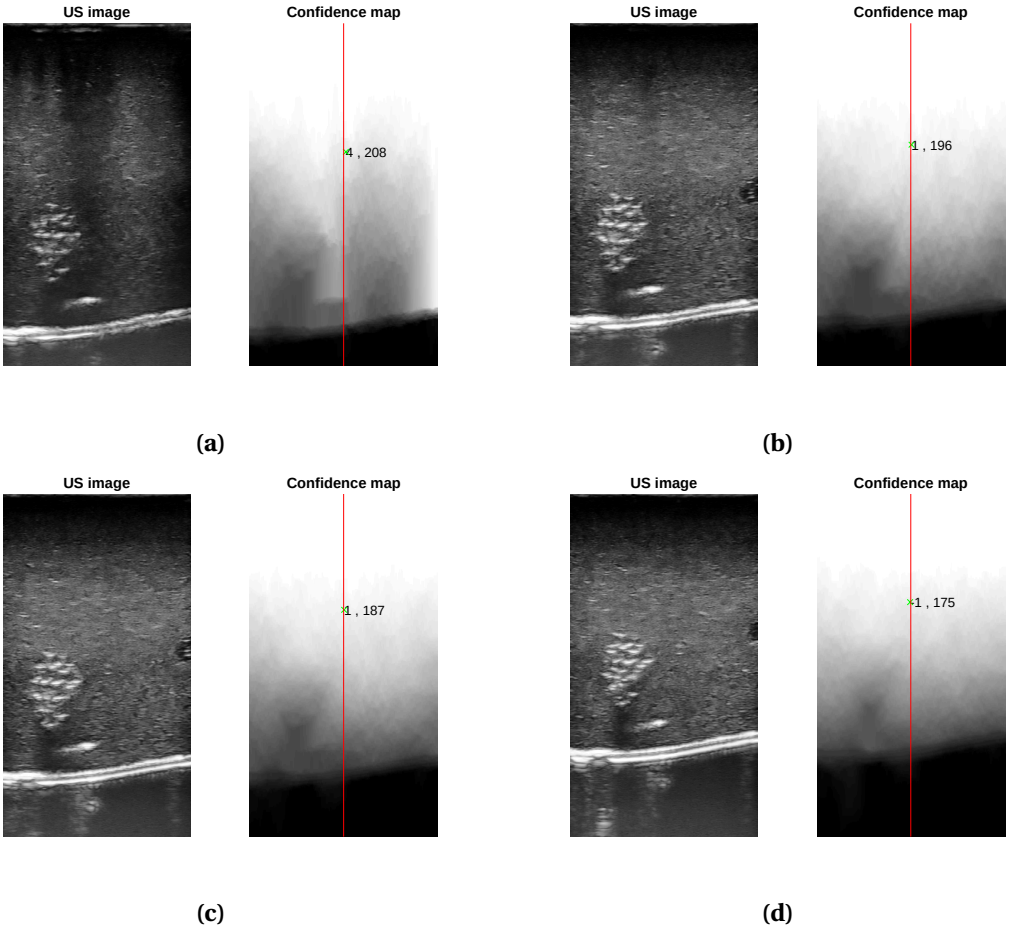


Figure 11.16: Ultrasound images and corresponding confidence maps for increasing contact pressure from **(a)** to **(d)**.

means that deeper tissue also appears in the image. In this experiment, the confidence is zero beneath a curved line in the bottom of the image, which can be a shadowing object such as a rib. The zero-confidence region increases when the contact pressure is increased, such that the confidence-weighted barycenter moves towards the top of the image. Beside, the position of the target object also changes due to the increasing contact pressure.

From this experiment we conclude that it is infeasible to find a generic relationship between the translation of the probe and the vertical displacement of the confidence-weighted barycenter. The composition of the imaged tissue and the presence of shadowing objects are factors that differ in every situation.

As an alternative we investigate the relation of the translation of the probe and the mean confidence of the image C_{mean}, which is a metric of global image quality. A similar setup is used as for the previous experiment, but now there is no contact between the probe and the phantom in the initial position. The global z-position of the ultrasound probe is then decreased such that the probe makes contact with the phantom and subsequently increases the contact pressure. Then the probe moves back to its initial position, such that the contact pressure decreases and finally the probe loses contact with the phantom. This motion is repeated five times while ultrasound images are captured at 30 frames per second. During the contact phase, the probe is always perpendicular to the surface of the phantom.

Results of this experiment are illustrated in figure 11.17. The top figure illustrates the mean confidence of the ultrasound images that are captured during the vertical motion that is

illustrated in the bottom figure. The horizontal black line in the bottom figure indicates the top surface of the phantom when no pressure is applied to it. The green stars indicate when the mean confidence is maximal in the downward motion. The figure shows a large increase/decrease when contact between the probe and the phantom is initiated/lost. Increasing the contact pressure initially leads to a relatively small increase in the mean confidence. Increasing the contact pressure even more decreases the mean confidence, but this can also be caused by the transition between the downward motion and upward motion.

From this experiment we conclude that increasing the contact pressure does not significantly improve the global image quality, while it can possibly cause discomfort to the patient. However, a loss of contact between the ultrasound probe and the phantom is unacceptable. Therefore we propose to set a threshold for the mean confidence to detect contact loss. When the mean confidence of an image drops below this threshold, the probe is commanded to move normal to the surface of the phantom with a value dz until the mean confidence exceeds the threshold. This control action will reposition the probe such that it regains contact with the phantom. According to figure 11.17 we suggest a threshold of C_{mean} = 0.6. By this approach we redirect our focus to contact detection, which has been demonstrated to be the more urgent factor to control for global image quality optimization.

An experiment was performed to validate our proposed method for contact loss between the ultrasound probe and the phantom. The initial pose of the probe was such that its contact surface was just above the surface of the phantom and the probe was oriented perpendicular to the surface of the phantom. In the initial position (Δz = 0 mm) there was no contact between the probe and the phantom, resulting in a confidence below the threshold value (figure 11.18).

The experiment was performed with a weighting factor w = 1 and a normal displacement dz = 1mm per control action. As illustrated in figure 11.18, it took 5 steps to exceed the threshold value of C_{mean} = 0.6, which is equal to a total normal displacement of Δz = 5 mm. The corresponding ultrasound images are illustrated in figure 11.19, which verify that there is contact after a normal displacement of 5 mm. These results validate that our proposed method detects and controls contact between the ultrasound probe and the phantom appropriately.

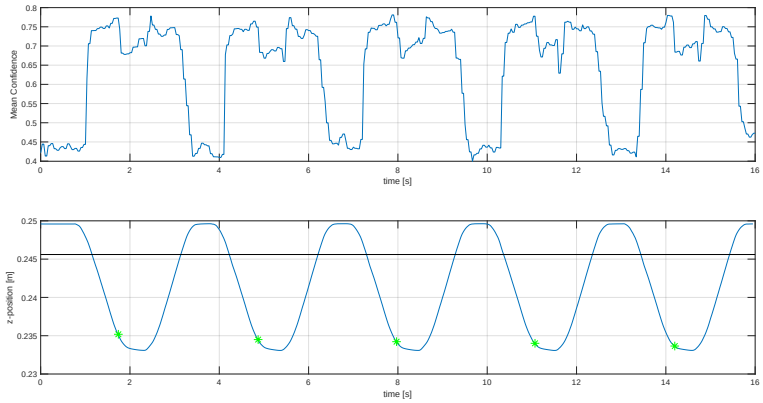


Figure 11.17: Top figure: mean confidence of the ultrasound images that are captured during the vertical motion. **Bottom figure:** Global z-position of the ultrasound probe. The horizontal black line indicates the top surface of the phantom when no pressure is applied to it. The green stars indicate when the mean confidence is maximal in the downward motion.

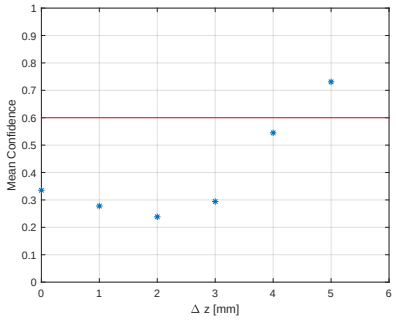


Figure 11.18: Mean confidence value C_{mean} for different values of Δz with respect to an initial position where the ultrasound probe is not in contact with the surface of the phantom.

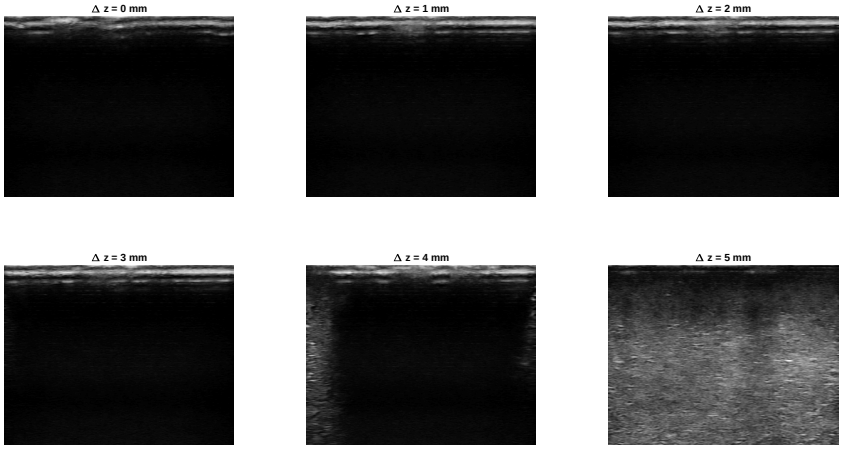


Figure 11.19: Ultrasound images during the transition between no contact and contact between the ultrasound probe and the phantom.

Discussion

Experiments have shown that it is (not yet) possible to derive a generic relation between the translation of the probe and the vertical displacement of the confidence-weighted barycenter. As an alternative, we used the main confidence of the image to detect contact loss between the ultrasound probe and the skin surface. This improves the image quality significantly when the planned trajectory deviates from the ideal trajectory at time of scanning. This feature is useful in clinical applications, where movements of the patients are inherent to the scanning procedure.

In-plane rotation

In this experiment we positioned the probe such that one part of the contact surface of the probe is not in contact with the phantom. The captured ultrasound image and corresponding confidence map are illustrated in figure 11.20. The in-plane rotation that results from the visual servoing algorithm, calculated by equation (6.14), is equal to $\theta_c = -0.0912 rad$. A weighting factor $w = 0$ was used, such that the control implements the control action just as it is suggested by the visual servoing algorithm. The ultrasound image and corresponding confidence map after the execution of the control action for in-plane rotation are illustrated in figure 11.21. The black stroke in the ultrasound image has disappeared, which means that the total contact surface of the probe is now in contact with the phantom. As a result, the barycenter of the

confidence map has moved closer to the central scan line. From this, we can conclude that the control action for in-plane rotation appropriately corrects for the horizontal position of the confidence-weighted barycenter.

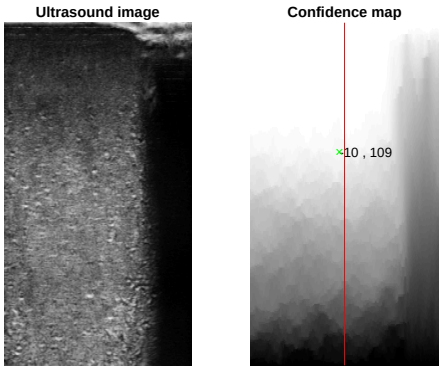


Figure 11.20: Ultrasound image and corresponding confidence map before the control action for in-plane rotation.

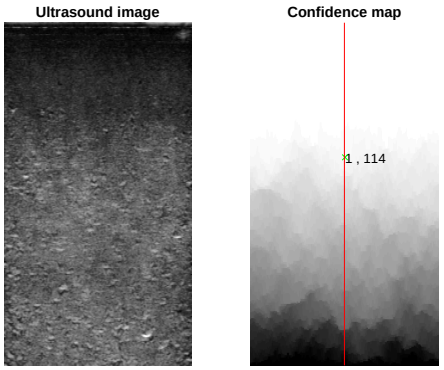


Figure 11.21: Ultrasound image and corresponding confidence map after the control action for in-plane rotation.

Discussion

From this experiment we conclude that the visual servoing algorithm is capable of optimizing the global image quality. The translation of the probe normal to the skin surface can be used to ensure contact between the ultrasound probe and the skin. The control action for in-plane rotation appropriately corrects for the horizontal position of the confidence-weighted barycenter and improves the confidence of the global image. By this, we proved that our control algorithm is capable to robotically manipulate an ultrasound probe in order to improve the global quality of the captured ultrasound images online by using a compliant control strategy that ensures safe human-robot interaction.

12 | Conclusion

The goal of this project was to develop a control strategy for autonomous ultrasound imaging that can be incorporated in a robotic biopsy procedure. The research objective of this study was defined as follows:

Design and implementation of a compliant control strategy for autonomous ultrasound scanning that establishes safe human-robot interaction.

To meet the research objective, we defined two milestones:

- Design and implementation of a compliant control strategy
- Design and implementation of an ultrasound-based visual servoing algorithm

We developed an energy-based impedance controller that incorporates safety limitations to establish safe human-robot interaction and energy tanks to guarantee the passivity of the overall system. Experiments have shown that the robot can follow a scanning trajectory with appropriate compliance, such that movements of the patient are absorbed. Clinicians use a similar compliant control of their arm to manipulate the probe in HHUS.

Deformations of the breast and movements of the patient yield abnormalities in the planned trajectory, which can result in an unsatisfying 3D image of the breast. A clinician uses a screen for visual feedback of the ultrasound images. We used a visual servoing algorithm based on confidence maps to improve the global image quality at the time of scanning. The visual servoing controls the in-plane rotation and translation of the probe perpendicular to the skin surface to correct for contact loss between the probe and the skin surface. In this way, visual feedback is provided to the robot which significantly improves the robustness of the scanning procedure.

A port-Hamiltonian model of the robot was created to test and evaluate the robot controller in simulation. Testing in simulation allows a faster design of the controller and reduces the risk of damage on the physical robot. Bond graphs were used as modelling tool, which assures conservation of energy in the interaction between different elements of the model. After validation of the controller design in simulation, the controller was implemented on the physical robot.

Experiments on the physcial robot have demonstrated a proof of concept of our proposed control strategy. This approach is very promising for autonomous ultrasound scanning, because it combines the advantages of both HHUS and robotic scanning. The controller equips the robot with a dynamic behaviour and visual information that resembles HHUS, while it takes advantage of the precision, computational power and insensitivity to fatigue of the robot. We believe that this contribution has made a big step towards the implementation of autonomous ultrasound scanning for clinical use.

12.1 Recommendations

To support the ongoing work on this project, we propose the following recommendations to further improve the proposed control strategy:

- **Automatic tuning of controller parameters:** The values of E_{max}, P_{max}, s, K_o, K_t, K_c and b are open parameters that are manually chosen in the current controller design. As demonstrated in the experiments, the behaviour of the controller changes significantly for a change in of the open parameters and the ideal settings depend on the task. Therefore, automatic tuning of the open parameters would make the control more generic for different tasks.
- **Automatic tuning of confidence parameters:** The same holds for the open confidence parameters α, β and γ as for the controller parameters. Automatic tuning of these parameters will make the control more generic for different types of tissue.
- **Continuous supervision of energy tanks:** The current implementation provides the energy tanks with a certain amount of energy at the beginning of a task and leaves it like that. This means that the energy tanks must be provided with sufficient energy to perform the task and hence they cannot account for large instantaneously energy drops within this range. On the other side, a task cannot be finished when the provided energy is too low. Therefore, continuous supervision of the energy tanks is proposed to provide the tank with sufficient energy to complete the short-term task, while the energy level is sufficiently low to account for undesirable energy drops.
- **Improvement of the ultrasound phantom:** The first prototype of the ultrasound phantom can be improved by decreasing the attenuation inside the phantom, such that it yields more anatomical detail in the ultrasound images. The use of other materials and contrast media can be investigated to decrease scattering and reflection between the different layers.
- **Creating a real-time system:** The main bottleneck regarding real-time-capability of the system is the computation of the confidence maps. We suggest to use sensory integration or Kalman filtering to only use visual information when it is available. This will speed up the scanning procedure, since the robot does not have to wait for the confidence estimation while executing the scanning trajectory.

A | Mathematical Definitions

In this appendix we will provide definitions of some mathematical concepts that are used in this report. An extensive explanation of these concepts is given in e.g. [67], [69], [42] and [26].

Definition 1. Mapping:
Given two sets X and Y, with $x \in X$, $y \in Y$ and $f(x) \in Y$, the mapping f from X to Y is denoted as:

$$f : X \to Y; \quad f(x) \mapsto y$$

Definition 2. Surjective, injective, bijective:
A mapping $f : X \to Y$ is *injective* (one-to-one) if $\forall x, x' \in X, f(x) = f(x') \Rightarrow x = x'$, it is *surjective* if $\forall y \in Y, \exists x \in X : y = f(x)$ and it is *bijective* if it is both injective and surjective.

Definition 3. Vector space:
An n-dimensional *vector space* V^n is a set of elements (called vectors) that contains a zero-vector $\vec{0}$ and two operations $\oplus$ (addition of two vectors) and $\cdot$ (multiplication of a vector by a scalar), such that:

- $\forall v_1, v_2 \in V$ also $v_1 \oplus v_2 \in V$
- $\forall v_1 \in V$ and $\forall x \in \mathbb{R}$ also $x \cdot v \in V$
- $\forall v_1 \in V \; \exists \; v^{-1} \in V$ such that $v_1 \oplus v_2 = \vec{0}$

Definition 4. Linear operator:
Given two vector spaces X and Y with $x \in X$ and $y \in Y$, then the mapping $A : X \to Y$ is said to be a linear operator if:

$$A(x + y) = Ax + Ay \tag{A.1}$$
$$A(\lambda x) = \lambda Ax \tag{A.2}$$

Definition 5. Projective space:
The *projective space* of dimension n, denoted as P^n, is the set of lines through the origin in $\mathbb{R}^{n+1}$.

Definition 6. Algebraic structure:
An *algebraic structure* is a pair (A, T) that consists of a set A and operation T, such that T is a map of the form:

$$T : A \times A \to A$$

Definition 7. Diffeomorphism:
Given two sets $\mathcal{M}$ and $\mathcal{N}$, a mapping $\phi : \mathcal{M} \to \mathcal{N}$ is a *diffeomorphism* iff it is bijective and the functions ϕ and ϕ^{-1} are differentiable.

Definition 8. Manifold:
A manifold $\mathscr{M}$ can be defined as a smooth and curved set which is locally diffeomorphic to $\mathbb{R}^n$ around each of its points.

Definition 9. Tangent space:
Given a manifold $\mathscr{M}$, we define the tangent space $T_{p(t)}\mathscr{M}$ to $\mathscr{M}$ at $p(t)$ as a smooth n-dimensional vector space that contains the vector $\dot{p}(t)$.

Definition 10. Group:
A *group* is an algebraic structure (A, T) that satisfies the following properties:

- Associativity: $a_1, a_2, a_3 \in A; (a_1 \cdot a_2) \cdot a_3 = a_1 \cdot (a_2 \cdot a_3)$
- Identity: $\exists I \in A : \forall a \in A \quad a \cdot I = I \cdot a = a$
- Inverse: $\forall a \in A \quad \exists a^{-1} \in A : a \cdot a^{-1} = I$

Definition 11. Lie group:
A *Lie group* is a group that is also a manifold.

Definition 12. Lie algebra:
A *Lie algebra* is the tangent space in the identity of a Lie group. It is a vector space V together with a bilineary operator $[.]$ called *Lie bracket*, that satisfies the following properties for $v_1, v_2, v_3 \in V$ and $\forall x_1, x_2 \in \mathbb{R}$:

- Bilinearity: $\begin{cases} [x_1 v_1 + x_2 v_2, v_3] & = x_1 [v_1, v_3] + x_2 [v_2, v_3] \\ [v_1, x_1 v_2 + x_2 v_3] & = x_1 [v_1, v_2] + x_2 [v_1, v_3] \end{cases}$
- Skew-symmetry: $[v_1, v_2] = -[v_2, v_1]$
- Jacobi's identity: $[v_1, [v_2, v_3]] + [v_2, [v_3, v_1]] + [v_3, [v_1, v_2]] = 0$

B | Bond graphs

Bond graphs offer a graphical representation of a physical system in such a way that conservation of energy is assured [42]. They consist of power bonds and elements. The power bonds represent the energy flow, i.e. the power, between the elements. The elements represent the physical concepts of energy storage, dissipation and transformation.

B.1 Power Bonds

The power bonds represent the power exchanged between two elements. They are defined by two dual variables, flow f and effort e, that together determine the power flow through the port. In case of a rotational joint of a serial manipulator, flow is the angular velocity and effort is the torque.

Power bonds are illustrated as half arrows, where the direction of the arrow represents the direction in which the energy flows. Bonds can be single or multi-dimensional, with respectively $(e, f) \in \mathbb{R}$ or $(e, f) \in \mathbb{R}^n$. Causal strokes are used to specify the relative direction of the effort. They are represented by a bold black line at the beginning or end of a power bond. If the causal stroke is at the beginning of the arrow, the effort goes in opposite direction as the energy. If the causal stroke is at the end of the arrow, the effort goes in the same direction as the effort. Two general bond graphs are represented in figure B.1.

Figure B.1: Single-bond with effort going from B to A (left) and multi-dimensional bond with effort going from A to B (right) [42].

B.2 Energy Storage Element

A physical storage element, like a spring or an inductor, is in a bond graph represented by an energy storage element. Power bonds connected to a storage element are always directed towards the storage element, which means that positive power is going in the direction of the storage element. Storage elements can be represented in integral form or differential form. In the integral form, the state of the element results from the integration of the input. In the differential form, the state of the system results from differentiation of the input. There are two types of storage elements. For single-bonds they are called **C** and **I** and for multi-dimensional bonds $\mathbb{C}$ and $\mathbb{I}$ are used.

A **C** storage element in its integral form has flow as input and effort as output. The state space equations are given by:

$$\begin{aligned} \dot{x}(t) &= f(t) \\ e(t) &= \gamma(x(t)) = \frac{\partial E}{\partial x}(t) \end{aligned} \tag{B.1}$$

With x the physical state of the system and *E(x)* an energy function of the state x. For a **C** element in integral form, the energy is called generalized potential energy.

A **C** element in differential form has variables in inverted direction compared to the integral form, meaning effort as input and flow as output. The inverse of γ from equation (B.1) is required to derive the state space equations. γ is a gradient of the energy function and a function of x. The inverse of γ will be a function of e, so it is no longer a function of the state. A new energy function, called co-energy *E*(e)*, is introduced to derive the state space equations (equation (B.2)). The **C** elements in integral form and differential form are illustrated in figure B.2.

$$\begin{aligned} x(t) &= \gamma^{-1}(e) = \frac{\partial E^*(e)}{\partial e} \\ f(t) &= \frac{dx}{dt}(t) \end{aligned} \tag{B.2}$$

Figure B.2: C element in integral form (left) and in differential form (right) [42].

An **I** storage element in its integral form has effort as input and flow as output. The state space equations are given by equation (B.3). For an **I** element in integral form, the energy is called generalized kinetic energy.

$$\begin{aligned} \dot{x}(t) &= e(t) \\ f(t) &= \gamma(x(t)) = \frac{\partial E}{\partial x}(t) \end{aligned} \tag{B.3}$$

An **I** element in differential form has again variables in inverted direction compared to the integral form, so flow as input and effort as output. The inverse of γ from equation (B.3) is again required to derive the state space equations. The inverse of γ will now be a function of f, so it is no longer a function of the state. So the co-energy *E*(e)* should be used again to derive the state space equations of equation (B.4). The **I** elements in integral form and differential form are represented in figure B.3.

$$\begin{aligned} x(t) &= \gamma^{-1}(f) = \frac{\partial E^*(f)}{\partial f} \\ e(t) &= \frac{dx}{dt}(t) \end{aligned} \tag{B.4}$$

Figure B.3: I element in integral form (left) and in differential form (right) [42].

B.3 Energy Dissipation Element

Energy dissipation elements represent physical systems that dissipate energy, like electrical resistors or mechanical dampers. Power bonds connected to a dissipation element are always directed toward the dissipation element, which means that positive power is going in the

direction of the dissipation element. The representation of a dissipation element is given in figure B.4.

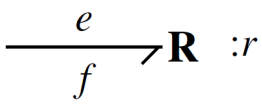


Figure B.4: Bond graph representation of a dissipative element [42].

An ideal dissipater is defined by a static relation between effort and flow, which can be given in impedance form Z or admittance form Y:

$$\begin{aligned} e &= Z(f) \\ f &= Y(e) \end{aligned} \tag{B.5}$$

To ensure that the energy can only flow towards the dissipative element, so no energy can be generated by the dissipative element, the following must hold:

$$Z(f)f \leq 0 \quad or \quad e(Y)e \leq 0 \tag{B.6}$$

B.4 Energy Transformation Elements

Energy transformation elements have two power bonds connected to them. They cannot store energy and are power continuous, which means that the energy flowing into the element is at every moment equal to the energy flowing out of the element. Physical examples are mechanical reducers and electrical transformers. The power-continuity is given by equation (B.7). The relation between the port variables determines whether the energy transformation element is a transformer or a gyrator.

$$P_{in} = e_{in}^T f_{in} = e_{out}^T f_{out} = e_{=} P_{out} \tag{B.7}$$

B.4.1 Transformers

A transformer element **TF** describes a linear relation between flows, and consequently also between efforts:

$$\begin{aligned} f_{out} &= n f_{in} \\ e_{out} &= \frac{1}{n} e_{in} \end{aligned} \tag{B.8}$$

Where n is a linear constant. A transformer element can be used to model e.g. a gear box, with n the reduction ratio. When n is variable, we speak of a modulated transformer **MTF**.

Energy transformation elements can be single or multi-dimensional. In the multi-dimensional case holds that $N \in \mathbb{R}^n \times \mathbb{R}^n$ and the transformer element and the modulated transformer element are given by respectively $\mathbb{TF}$ and $\mathbb{MTF}$. The bond graph representation of all transformer elements are given in figure B.5.

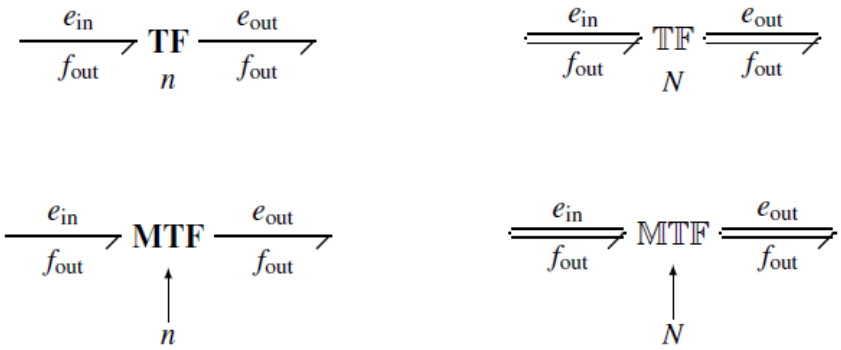


Figure B.5: Bond graph representation of transformer elements [42].

B.4.2 Gyrators

A gyrator **GY** describes a linear relation between output effort and input flow, and consequently output flow and input effort as:

$$\begin{aligned} e_{out} &= n f_{in} \\ f_{out} &= \frac{1}{n} e_{in} \end{aligned} \tag{B.9}$$

A physical example of a gyrator is a DC motor, in which current (effort) is transformed into torque (flow). If n is variable, we speak of a modulated gyrator **MGY**. In the multi-dimensional case, the gyrator element and modulated gyrator element are given by respectively $\mathbb{GY}$ and $\mathbb{MGY}$. The bond graph representation of all gyrator elements is given in figure B.6.

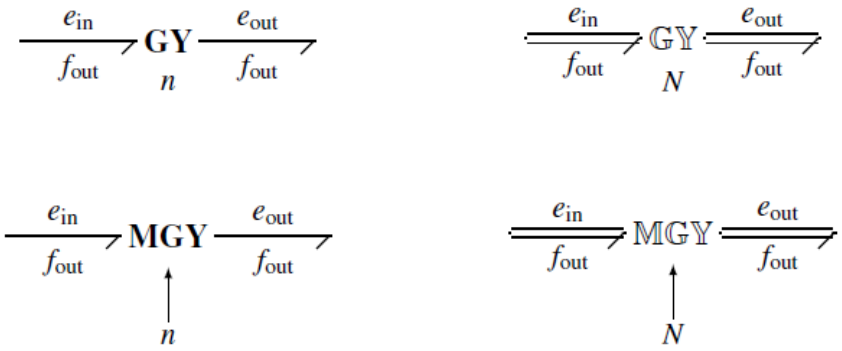


Figure B.6: Bond graph representation of gyrator elements [42].

B.4.3 Energy source elements

There are two types of energy source elements. An effort source S_e (or $\mathbb{S}_e$ in the multi-dimensional case) can supply effort independently of the dual flow. On the other hand, a flow source S_f (or $\mathbb{S}_f$ in the multi-dimensional case) can supply flow independently of the dual effort. Since energy source elements provide energy to the system, the power bonds connected to them are always directed away from the element. The bond graph representation of both energy source elements is given in figure B.7.

Figure B.7: Bond graph representation of an effort source element (left) and a flow source element (right) [42].

B.4.4 Junction elements

Junction elements are power continuous elements, that can have any number of bonds attached to them. They provide an interconnection between all the other types of elements. There are two types of junctions: **0**-junction and **1**-junction. The bond graph representation of both is given in figure B.8.

A **0**-junction, or effort junction, has bonds with the same effort value connected to it. One of the bonds sets the effort value, while the other bonds use it. Since the junction element is power continuous, the sum of flows going into the element is equal to the sum of flows going out.

A **1**-junction, or flow junction, has bonds with the same flow value connected to it. One of the bonds sets the flow value, while the other bonds use it. For the **1**-junction element, the sum of efforts going into the element is equal to the sum of efforts going out.

Figure B.8: Bond graph representation of a **0**-junction (left) and a **1**-junction (right) [42].

C | 20-sim

The 20-sim software is a modeling and simulation program that allows modeling by equations, block diagrams, physical components and bond graphs [1]. The 20-sim editor is illustrated in figure C.1. A model can be built from different submodels, e.g. a controller and a model of a physical system. Every model is organized in a certain hierarchy, where the highest level represents the total system and the lowest level contains only equations. For example, different levels in the model of a physical robot can be: the robot, a joint, an energy transformation element, and the equations describing the energy transformation element.

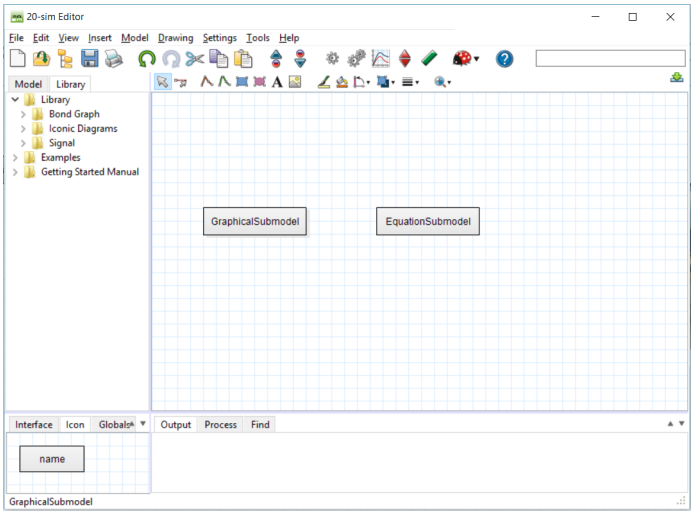


Figure C.1: 20-sim editor

There are basically two types of submodels: graphical submodels and equation submodels. A graphical submodel allows to drag and drop elements (e.g. bond graph elements, block diagrams and physical components) from a library. These elements can be connected by signals or power ports. A graphical submodel of a robot joint is given in figure 7.6. A half arrow represents a power port and a full arrow represents a signal. A single arrow is used for one-dimensional cases and a double arrow for multi-dimensional cases.

As can be seen in figure 7.6, a graphical submodel can consist of library elements and customized building blocks. A customized building block adds another level of hierarchy into the model. This can be an equation submodel or again a graphical submodel. The lowest level of the model is always an equation submodel and it describes the relation between the different elements in the form of equations. Also the elements itself are described by equations. The standard format of an equation submodel is given in figure C.2.

When the model is finished, a simulation can be run that allows graphical representation of different variables in the system. It is also possible to create a 3D animation of the system.

```
/* Equation Submodel
Enter your equations here.
*/
parameters
   real global x[7];
   real y[3];
variables
   real z = x[4];
   real a;
equations
   a = z*x[2];
```

Figure C.2: Format of an equation submodel.

Bibliography

[1] Home page 20-sim. http://www.20sim.com/. Accessed: 25-04-2018.

[2] Home page ces edupack. http://www.grantadesign.com/education/edupack/. Accessed: 16-05-2018.

[3] Home page murab project website. http://www.murabproject.eu/. Accessed: 24-04-2018.

[4] Magewell usb capture hdmi gen 2. http://www.magewell.com/products/usb-capture-hdmi-gen-2. Accessed: 27-06-2018.

[5] Models of friction. http://www.mogi.bme.hu/TAMOP/robot_applications/ch07.html#ch-8.3.2. Accessed: 30-04-2018.

[6] Point spread function (psf). https://www.zeiss.com/microscopy/us/solutions/reference/basic-microscopy/the-point-spread-function.html. Accessed: 24-05-2018.

[7] Siemens acuson p500 ultrasound system. https://www.healthcare.siemens.nl/ultrasound/ultrasound-point-of-care/acuson-p500-ultrasound-machine. Accessed: 15-06-2018.

[8] Siemens l10-5v linear transducer. https://static.healthcare.siemens.com/siemens_hwem-hwem_ssxa_websites-context-root/wcm/idc/groups/public/@global/@imaging/@ultrasound/documents/download/mda2/ntc2/~edisp/us_acuson_nx2_transducer_flyer-03577048.pdf. Accessed: 15-06-2018.

[9] Telemed ultrasound database. http://www.telemedultrasound.com/?lang=en.

[10] Ultimaker 2+ product page. https://ultimaker.com/en/products/ultimaker-2-plus. Accessed: 24-04-2018.

[11] Ultrasound production and interactions. http://www.sprawls.org/ppmi2/USPRO/. Accessed: 24-05-2018.

[12] United states breast cancer statistics. http://www.breastcancer.org/symptoms/understand_bc/statistics. Accessed: 18-04-2018.

[13] Y. Y. An, S. H. Kim, B. J. Kang, and J. H. Lee. Usefulness of magnetic resonance imaging-guided vacuum-assisted breast biopsy in Korean women: a pilot study. 2013.

[14] N. Arora, T. A. King, L. M. Jacks, M. M. Stempel, S. Patil, E. Morris, and M. Morrow. Impact of breast density on the presenting features of malignancy. 2010.

[15] R. Ball. *Theory of screws*. 1900.

[16] J. Bao, P. L. Lee, and B. E. Ydstie. *Process Control: The Passive Systems Approach*. 2010.

[17] A. Cafarelli, P. Miloro, A. Verbeni, M. Carbone, and A. Menciassi. Speed of sound in rubber-based materials for ultrasonic phantoms. 2016.

[18] C. Cardenas. Development of a Safety-Aware Intrinsically Passive Controller for Multi-DOF Manipulator. 2017.

[19] P. Chatelain, A. Krupa, and N. Navab. Optimization of ultrasound image quality via visual servoing. 2015.

[20] P. Chatelain, A. Krupa, and N. Navab. Confidence-Driven Control of an Ultrasound Probe. 2017.

[21] F. Chaumette and S. Hutchinson. Visual servo control. Part I: Basic approaches.

[22] Y. H. Chou, C. M. Tiu, J. Chen, and R. F. Chang. Automated full-field breast ultrasonography: The past and the present, 2007.

[23] F. Conti, J. Park, and O. Khatib. Interface Design and Control Strategies for a Robot Assisted Ultrasonic Examination System. 2010.

[24] S. de Looijer and M. Welleweerd. Control strategies for autonomous medical ultrasound imaging. 2018.

[25] D. Dresscher, Y. Brodskiy, J. F. Broenink, S. Stramigioli, Y. Brodskiy, P. Breedveld, and J. Broenink. Modeling of the youBot in a serial link structure using twists and wrenches in a bond graph.

[26] V. Duindam, A. Maccheli, S. Stramigioli, and H. Bruyninckx. *Modeling and Control of Complex Physical Systems*. 2009.

[27] Y. Erez, Y. Y. Schechner, and D. Adam. Ultrasound Image Denoising by Spatially Varying Frequency Compounding. 2006.

[28] P. Fitts. Human engineering for an effective air-navigation and traffic-control system. 1951.

[29] G. A. Folkertsma and S. Stramigioli. Energy in Robotics. 2017.

[30] L. Grady. Random Walks for Image Segmentation. 2006.

[31] B. C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introuction*. 2015.

[32] N. Hogan. Impedance Control: An Approach to Manipulation. 1985.

[33] B. Ihnatsenka and A. P. Boezaart. Ultrasound: Basic understanding and learning the language. 2010.

[34] Y. Jae, K. Jong, H. Seo, H. Rae, K. Kwang, and G. Kim. Development of a control algorithm for the ultrasound scanning robot (NCCUSR) using ultrasound image and force feedback. 2016.

[35] M. Kanat, C. A. Agung, J. Ramkrishna, P. Jacquelien, and J. Scherpen. *Mathematical Control Theory I: Nonlinear and Hybrid Control Systems*. 2015.

[36] A. Karamalis, W. Wein, T. Klein, and N. Navab. Ultrasound confidence maps using random walks. 2012.

[37] K. M. Kelly, J. Dean, W. S. Comulada, and S. J. Lee. Breast cancer detection using automated whole breast ultrasound and mammography in radiographically dense breasts. 2010.

[38] K. M. Kelly, J. Dean, S. J. Lee, and W. S. Comulada. Breast cancer detection: Radiologists' performance using mammography with and without automated whole-breast ultrasound. 2010.

[39] K. M. Kelly and G. A. Richwald. Automated Whole-Breast Ultrasound: Advancing the Performance of Breast Cancer Screening. 2011.

[40] KUKA Roboter GmbH. KUKA FRI Documentation. 2016.

[41] KUKA Roboter GmbH. KUKA Sunrise Documentation. 2017.

[42] M. Laghi. *Passive based control on a KUKA arm*. PhD thesis, 2014.

[43] L. Leger, R. Guardo, P. Montréal, and B. G. Pike. Adipose tissue volume measured by magnetic resonance imaging and computerized tomography in rats. 1991.

[44] C. P. Loizou, C. S. Pattichis, M. Pantziaris, T. Tyllis, and A. Nicolaides. Quality evaluation of ultrasound imaging in the carotid artery based on normalization and speckle reduction filtering. 2006.

[45] E. Luper. *Component-Based Modelling and Simulation of a KUKA LWR+4 Robotic arm*. PhD thesis, 2017.

[46] L. E. Maggi, M. A. Von Krüger, W. C. A. Pereira, and E. E. C. Monteiro. Development of silicon-based materials for ultrasound biological phantoms. 2009.

[47] R. Mebarki, A. Krupa, and F. Chaumette. 2-D ultrasound probe complete guidance by visual servoing using image moments. 2010.

[48] T. Moller and B. Trumbore. Fast, minimum storage ray-triangle intersection. 1998.

[49] G. Mozzi. *Discorso matematico sopra il rotamento momentaneo dei corpi*. 1763.

[50] C. Nadeau, A. Krupa, J. Petr, and C. Barillot. Moments-based ultrasound visual servoing: From a mono- to multiplane approach. 2016.

[51] C. Nadeau, H. Ren, A. Krupa, and P. Dupont. Intensity-based visual servoing for instrument and tissue tracking in 3D ultrasound volumes. 2015.

[52] A. Ng and J. Swanevelder. Resolution in ultrasound imaging. 2011.

[53] R. Ortega, A. Loriat, R. Kelly, and L. Praly. On passivity-based output feedback global stabilization of Euler-Lagrange systems. 1994.

[54] S. Parker, W. Jobe, A. Dennis, MA andStavros, and K. Johnson. US-guided automated large-core breast biopsy. 1993.

[55] H. Paynter. Analysis and design of engineering systems. 1960.

[56] A. C. Pereira. Automated breast ultrasound: Current state-of-the-art and future aspects. 2013.

[57] H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano. Task-space control of robot manipulators with null-space compliance. 2014.

[58] H. G. Sage, M. F. De Mathelin, and E. Ostertag. Robust control of robot manipulators: A survey. 1999.

[59] P. E. Sandin. Robot Mechanisms and Mechanical Devices. 2003.

[60] M. Scarlett. Ultrasound-based Visual Servoing. 2015.

[61] J. Schoot Uiterkamp. Design and implementation of autonomous robotic scanning of the breast. 2016.

[62] C. Secchi, S. Stramigioli, and C. Fantuzzi. Control of Interactive Robotic Interfaces: A Port-Hamiltonian Approach. 2007.

[63] T. Shiferaw Tadele, T. de Vries, B. Siciliano, P. Jonker, and J. van Amerongen. Human-Friendly Robotic Manipulators: Safety and Performance Issues in Controller Design. 2014.

[64] B. Siciliano and J.-J. E. Slotine. A General Framework for Managing Multiple Tasks in Highly Redundant Robotic Systems. 1991.

[65] M. W. Spong, S. Hutchinson, and M. Vidyasagar. Robot Modeling and Control. 2006.

[66] S. Stramigioli. Creating artificial damping by means of damping injection. 1996.

[67] S. Stramigioli. *Modeling and IPC Control of Interactive Mechanical Systems: A Coordinate-Free Approach*. 2001.

[68] S. Stramigioli. *Mathematical Control Theory I*. 2015.

[69] S. Stramigioli. Lecture notes in modern robotics, 2017.

[70] S. Stramigioli and H. Bruyninckx. Geometry and Screw Theory for Robotics. 2001.

[71] S. Stramigioli and H. Bruyninckx. Geometry of dynamic and higher-order kinematic screws. 2001.

[72] S. Stramigioli, B. Maschke, and A. Van Der Schaft. Passive output feedback and port interconnection. 1998.

[73] T. S. Tadele, T. J. De Vries, and S. Stramigioli. Combining energy and power based safety metrics in controller design for domestic robots. 2014.

[74] G. Tarantino. Design and Implementation of automated breast scanning with force feedback control. 2017.

[75] G. E. Trahey, S. W. Smith, and O. T. Von Ra. Speckle Pattern Correlation with Lateral Aperture Translation: Experimental Results and Implications for Spatial Compounding. 1986.

[76] K. Ustuner and G. L Holley. Ultrasound imaging system performance assessment. 2018.

[77] A. Van Der Schaft. Port-Hamiltonian systems: an introductory survey. 2006.

[78] H. Y. Wang, Y. X. Jiang, Q. L. Zhu, J. Zhang, Q. Dai, H. Liu, X. J. Lai, and Q. Sun. Differentiation of benign and malignant breast lesions: A comparison between automatically generated breast volume scans and handheld ultrasound examinations. 2012.

[79] J. C. Willems. Dissipative Dynamical Systems Part 1: General Theory. 1972.

[80] S. Zintsmaster, J. Morrison, S. Sharman, and B. A. Shah. Differences in pain perceptions between automated breast ultrasound and digital screening mammography. 2013.