Mateusz Mesek

The noninvasive technique of determining local stiffness of human arteries

Doctoral dissertation

Supervisors:
Ziemowit Ostrowski, PhD, DSc
Prof. Leif Rune Hellevik

Science discipline:
Biomedical Engineering

This research was financed from
the Norwegian Financial Mechanism 2014-2021 within project ENTHRAL
"Non-invasive in-vivo assessment of stiffness of human artery walls"
under grant# UMO-2019/34/H/ST8/00624 (Norway Grants).

Silesian University of Technology
Norwegian University of Science and Technology
Gliwice, 2024

Author
Mateusz Mesek
Silesian University of Technology
Faculty of Energy and Environmeental Engineering
Department of Thermal Technology
Konarskiego St. 22
44-100 Gliwice
Poland

Norwegian University of Science and Technonlogy
Faculty of Engineering
Department of Structural Engineering
Richard Birkelands vei 1A
7034 Trondheim
Norway

e-mail: mateusz.mesek@polsl.pl; mateuslm@stud.ntnu.no;
mesek.mateusz@gmail.com

Contents

Chapter 1

Introduction

Cardiovascular diseases are the major cause of death around the world. It is estimated that 20% of the population is affected by elevated arterial wall stiffness which increases the risk of aneurysms rupture. Diminished arterial elasticity results in reduced pulse wave reflection and can lead to left ventricular hypertrophy or an increase in hemodynamic pressure gradient causing an increased cardiac load.

The stiffening of the vasculature, primarily results from aging as large elastic arteries undergo progressive luminal dilatation, thickening of the arterial wall, increased deposition of collagen, and combined fragmentation and degeneration of elastin fibers [1]. Additional contributions from diabetes, renal diseases, and other diseases are associated with coronary artery diseases, systolic hypertension, stroke, atrial fibrillation, and many other conditions [2, 3, 4].

In the normal physiological state, the elastic nature of the arteries ensures that they easily expand to accommodate the volume of blood rapidly ejected by the heart during systole (typically only a fraction of the time between heartbeats). In addition to reducing the force required to pump the blood out of the heart, the distension of the arteries stores the cardiac work as elastic energy. The elasticity of the arteries therefore buffers the pulsatility of the flow and pressure generated by the heart and sustains a nearly steady flow of blood to the periphery. It also shields the small vascular beds from sharp pressure gradients. Thus, in the pathophysiological case of significantly stiffened arteries, the work required to eject blood from the heart increases, which can thus stimulate left ventricular hypertrophy and left ventricular remodeling [2]. Furthermore,

a stiffened vasculature has a decreased buffering capacity and allows steep and amplified forward pressure waves to penetrate into smaller vascular beds in the kidneys and brain, promoting their damage [5].

Assessment of the stiffness of the arterial system is a valuable diagnostic index and pathophysiological marker [6, 7, 8]. Recent studies continue to demonstrate substantial prognostic value of aortic stiffness as a predictor of cardiovascular morbidity and mortality. Meta-analysis and recent reviews highlight the increasing recognition of stiffness assessment as a surrogate endpoint for cardiovascular disease. Therefore, the parameters that describe the wall stiffness and pressure gradient are used as indicators that help to diagnose the severity of the disease [9]. The stiffness may be assessed by measuring the pulse wave velocity (PWV) [10], shear wave elastography [11] or by solving an inverse problem to estimate unknown parameters of the constitutive law suitable for the artery wall [12].

Van Disseldorp et al. [12] assume constant thickness of the aortic wall which highly influences the value of stiffness. Even a slight change of order 10^{-3} mm changes the solution of the inverse problem. This thesis addresses the issue of thickness measurement leading to more accurate estimation of Young's modulus. The authors point out the need for verification of the aortic stiffness by *ex-vivo* bi-axial testing of the tissue. Only patients undergoing surgery would yield samples for *ex-vivo* testing. Similarly to Van Daisseldorp, the artery wall displacement is registered with ultrasound, however to check accuracy of the proposed inverse problem methodology in this work, artery phantom undergoes uni-axial tensile test after the displacement and pressure measurement.

The aim of the thesis is to develop a non-invasive procedure using ultrasound and a numerical model to inversely estimate Young's modulus of all the ultrasound accessible arteries.

1.1 Objectives

This work aims to develop a non-invasive procedure to estimate Young's Modulus of Left Common Carotid Artery (LCCA) based on ultrasound measurement of the artery's wall displacement and pressure measurement inside the artery using applanation tonometry.

The measurements, experimental set-up as well as the data analysis and computations were performed within "ENTHRAL Non-invasive *in-vivo* assessment

of stiffness of human artery walls" project financed from the Norwegian Financial Mechanism 2014-2021, under grant# UMO-2019/34/H/ST8/00624 (Norway Grants).

In order to achieve the goal of this thesis, a testing rig was built consisting of artery's phantom, pulsatile pump, and measurement devices to collect flow, pressure, and phantom's wall displacement data. The numerical model of the experiment is built and validated against the experimental data. The model is then used for the inverse problem - estimation of the stiffness defined here as Young's modulus of the material used as a substitute of the real artery. The constitutive law choice is restricted by the number of parameters to estimate. For this reason only two models are being considered - linear elastic and Neo-Hookean. The best model based on experimental rig's data is chosen for inverse problem solved by extended Kalman filter. This approach for Young's modulus estimation along with chosen constitutive law is then used for patient's data. Chapter 5 covers model validation, simulation results of linear elastic and Neo-Hookean materials are compared with experimental data, and the most appropriate model is chosen for further analysis in the next chapters.

1.2 Outline

The thesis is organized into 8 chapters.

Chapter 1 provides a literature review on existing techniques for non-invasive estimation methods and a literature review on the proposed method of solution of the inverse problem.

Chapter 2 introduces elements of continuum mechanics necessary to understand the formulation of constitutive models of linear elastic and hyperelastic materials. It covers inverse problems and the most commonly used techniques to solve these problems and Kalman filtering algorithms.

Chapter 4 is devoted to mesh sensitivity study, comparison of static and dynamic finite element analysis. It provides the details of the geometry, mechanical properties and describes the algorithm used to process the data.

Chapter 3 discusses the experimental set-up built to collect the data for developing and testing new methods for non-invasive assessment of the common carotid artery.

Chapter 6 provides an example based on semi-empirical data for verifying the

extended Kalman filter code and procedure. The geometry and Young's modulus were assumed to be known and the convergence of the filter was tested on exactly known data. In this chapter results for the experimental rig and patients are reported.

Chapter 7 is devoted to the discussion and conclusions. It offers potential improvements to the proposed approach in order to implement it in real-world applications.

Chapter 8 consists of additional materials, visualization of the working principle of Kalman filter and Python code for data processing and implementation of the Kalman filter.

1.3 Literature review

Young's modulus of the artery is linked to the Pulse Wave Velocity (PWV) by the Moens-Korteweg equation [13, 14]. The Pulse Wave Velocity is defined as the velocity at which pressure perturbations propagate along the vessel and serves as an indirect measure of the average stiffness of the arteries within the segment between two measurement points [7].

Typically, measurements are taken between the carotid and femoral arteries abbreviated as cfPWV and known as a golden standard. Other segments that are investigated are heart-femoral measurements resulting in hfPWV or brachial-ankle abbreviated as baPWV [10]. PWV is the technique most commonly used to characterize arterial stiffness and has been established as a statistically reliable prognostic indicator. The measurement of local PWV relies on the measurement of the time difference between the two characteristic points of the cardiac cycle (for example, systolic pressure), at two points whose distance along the vessel is known. The ratio of these differences gives the approximate velocity of the pressure wave [15, 16].

For the pulse wave velocity, an applatation tonometer is widely used [17]. This method provides the average velocity over a long distance suffering from inaccuracies due to challenges in measuring vascular length [18]. In addition, PWV varies significantly along the entire arterial tree due to variations in geometry and elastic properties. Pathologies originate from localized arterial stiffening and local methods are more important to assess the state of health [19].

Shear wave elastography allows direct, non-invasive, and real-time measurement

of the elasticity of soft tissues. Shear waves are generated by focused acoustic radiation force emitted from a linear ultrasound probe, which itself provides local stress and generates local displacement in the tissue [11]. Acoustic radiation forces is a period-averaged force exerted on the medium by a sound wave and it is produced by the change of energy of the propagating wave [20]. The propagation of the shear wave is captured by fast ultrasound imaging. Shear wave velocity c_s distribution at each pixel is related to the shear modulus from which Young's modulus is derived. The shear modulus is related to the shear wave velocity by $G = \rho c_s$. Where ρ is the density of the investigated tissue, usually approximated based on the literature study, are approximated as a value that is close to the density of water. The vessel is assumed to be incompressible $\nu = 0.5$, resulting in a simple relationship $E = 3G$ [18, 11].

Other ways of estimating the unknown stiffness rely on running the simulations of the mathematical model describing the material's constitutive law until the unknown parameter is retrieved [21, 22, 23]. The parameter estimation problem may be solved by a variational or sequential approach. The sequential approach is iterative in nature and requires many simulations of the forward problem, which may be prohibitive.

With a sequential approach based on the Kalman Filter, the model prediction is improved at every time step by measuring the discrepancy between model output and measurements. The total computational time for the sequential approach was shown to be of the same order of magnitude as the CPU time needed for one forward simulation [24].

1.3.1 Kalman filtering in previous studies

A scoping review was performed using multiple keyword combinations. The databases included for the literature search were ScienceDirect, Web of Science and Scopus. For snowballing Google Scholar was used. Combination of keywords consisted of Kalman filter, artery, stiffness, constitutive and soft tissue. The search was limited to research papers relevant to hemodynamic modeling and estimation of material constitutive law's parameters. The articles were first selected by title and then by abstract to fit the research questions. Next step in the screening process was skimming through the paper in order to find the methodology used for simulations. The papers which focused on incorporating Kalman Filter into the Finite Element Method were discarded. Some papers only mentioned about the Kalman Filter and the filter wasn't applied in these papers, these papers were removed as well.

Initial number of found papers during search reduced from 150 to 18 upon after screening. Tab.1.1 summarizes the results of the search and includes a detailed string used for the search.

Keyword string	**Science Direct**	**Web of Science**	**Scopus**
"Kalman filter" AND "artery" AND "stiffness"	49	5	7
"Kalman filter" AND "artery" AND "constitutive"	47	0	0
"Kalman filter" AND "soft tissue" AND "stiffness"	39	2	1
Number of identified papers	135	7	8
Number of papers after screening	11	4	2
Snowballing	1		
Included papers	18		

Table 1.1: Process of literature review and results summary The screening process was based on title and abstract screening during search in databases. Then the papers with irrelevant content were excluded.

Tab.1.2 shows the models that were analyzed in a particular study and methods which were used to estimate unknown parameters. Figure 1.1 summarizes this table by aggregating variations of filters such as Reduced Order Kalman Filter (ROKF) or Simplified Kalman Filter (SKF) into one group called 'KF based'. The same was done for variations of the Unscented Kalman Filter (UKF). Based on Fig.1.1 and tables Tab.1.2 and Tab. 1.3 an overview of the most frequently used methods is presented.

Habibi [26] used Dynamic Mode Decomposition to reduce the computational cost of 3D fluid flow simulations and provide a recursive linear dynamical model. They tested the ROM-KF framework for three cases: Wommersley's analytical solution (1D), idealized 2D cerebral aneurysm model and patient-specific cerebral aneurysm model to reconstruct the velocity profile inside a vessel. It was challenging however to accurately reconstruct wall shear stress (WSS) throughout the whole heart cycle. Coupling the reduced order model with KF is capable of reducing the reconstruction error in the near-wall region and pro-

Author	k	ν	WK	Other	Model	Method
Delalleau, A[25]	✓	✓			FEM (linear elasticity)	EKF
Habibi, Milad[26]				✓	1D Wommersley, 2D and 3D pulsatile flow	ROKF
Jonášová, Alena[27]				✓	0D and 3D	UKF
Canuto, Daniel[28]	✓		✓		0D, 1D, coupled 0D-1D	EnKF
Caiazzo, Alfonso[29]	✓				1D	ROUKF
Müller, Lucas O[30]	✓		✓		1D	ROUKF
Lombardi[31]	✓				1D	UKF
Jain, Karan[32]	✓		✓		1D	UKF
Bertoglio, Cristóbal[33]	✓				3D FSI	ROUKF
Rüschen, Daniel[34]				✓	1D	KF, EKF
Hullender, David A[35]				✓	Fourier series	EKF
Shi, Pengcheng[36]	✓	✓			FEM	KF, EKF
Jin, Ji[37]				✓	third order ODE	KF

Table 1.2: Overview of the methods, parameters and models analyzed in papers. Where k represents stiffness in general, ν - Poisson's ratio, WK - Windkessel parameters. Continued on next page.

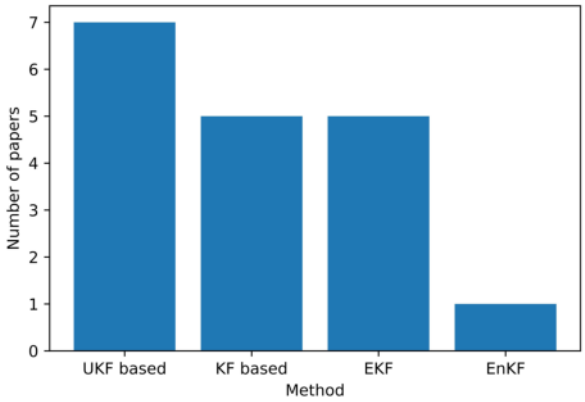


Figure 1.1: Bar plot representing the frequency of the specific method used in the selected papers. The methods are lumped into groups based on the underlying filter.

duces near-wall flow results more accurate than the uncertain computational data. Increasing noise in the experimental data to 0.3 of the maximum velocity as standard deviation reduces the accuracy of reconstructed experimental data during early and late stages of the cardiac cycle.

Ruschen reported in his work use of KF and EKF to estimate cardiac output by combining experimental measurements on animals and Windkessel model. They showed that KF underestimates the cardiac output and EKF is more accurate and shows no systematic deviation [34].

Pengcheng developed a stochastic finite element framework to estimate myocardial kinematics and material properties (Young's modulus, Poisson's ratio) and MRI data of canine tissue [36].

Ji Jing [37] introduced new quantification parameters to display the pressure waveform change caused by superimposition of wave reflection in the systolic reflex period. Kalman filter was used to obtain an optimal estimation of the new parameters and other parameters also describing the pressure waveform such as early systolic period, diastolic period and pulse pressure. He also showed that the new parameters correlate to commonly used wave reflection parameter AIx and contain more information than AIx.

Zhang used KF to estimate the external loads on the myocardium and its motion from magnetic resonance image sequence [38].

Author	k	ν	WK	Other	Model	Method
Zhang, Heye[38]				✓	2D FEM	KF
Gao, Zhifan[39]				✓	Newton's second law	KF
Liu, Wenyan[40]				✓	convolution of pressure waveforms	SKF
Bertoglio, Cristóbal[41]	✓				FSI	ROUKF
Song, Jialu[42]	✓				FEM	EKF

Table 1.3: Overview of the methods, parameters and models analyzed in papers. Where k represents stiffness in general, ν - Poisson's ratio, WK - Windkessel parameters. Continued.

Zhifan Gao applied Kalman filter to predict the radial and longitudinal motion of the carotid artery from ultrasound images. It was shown that the approach has the same order of accuracy as the manual tracing method performed by the medical physicians, and can hence be used to used to automate the motion tracking of the carotid artery wall [39].

Another application of linear Kalman Filter reported by Wenyan Liu et al. was non-invasive estimation of the aortic pressure waveform which is important for the diagnosis of cardiovascular diseases. As stated by the authors the proposed framework "outperforms the canonical correlation analysis (CCA), which is the current (by 2022) state-of-art blind system identification method for the non-invasive estimation of central aortic blood pressure" [40].

Delalleau [25] modified the EKF by introducing parameter λ for fading memory formulation of EKF. The parameter was inserted into the calculation of Kalman gain K and posterior covariance P. The proposed algorithm was tested on artificial data simulated by finite element method software SYSTUS released in 2004. Skin fragment was modelled as linear elastic solid and the force/displacement curve for uniaxial tensile test was registered. It was reported that the EKF is able to predict the Young's modulus and Poisson's ratio with high accuracy (mean error of order 10^{-9}).

Hullender showed the application of EKF to assess the blood pressure waveform in patients with pathologies that include hypertension and arterial stiffness. The pressure waveform was characterized by eleven-coefficient Fourier series with unknown frequencies and amplitudes [35].

Jialu Song [42] combined maximum likelihood theory, EKF and nonlinear Finite Element Method to identify unknown hyperelastic material (Neo-Hookean) constants for human liver. The methodology was verified with experimental analysis. 10x10x10 mm cube made of silicone gel with similar mechanical properties was subjected to an indentation test. Vertical time-varying force was applied at the center of the sample's top surface by a rigid needle. Two nodes at the top surface were selected in the computational model to compare their displacements in the x and y directions with experimental data.

Jonasova applied UKF to the coupled 0D three-element Windkessel model with 3D CFD simulation to find the best estimates of input parameters to the 0D model for stenosed artery. The algorithm was demonstrated on two different patient-specific carotid bifurcations by including data from *in vivo* measurements provided by medical doctors [27].

Caiazzo [29] presented the UKF based on a reduced order model in the context of 1D blood flow models. Experimental flow and pressure measurements of the arterial network consisting of 37 silicone tubes were used to estimate Young's modulus and wall thickness using at most a single pressure or flow measurement per vessel. The uncertainty of initial conditions were neglected resulting in lowering the dimension of matrices to the size of the unknown parameter space. The estimation of these parameters was conducted in three ways: only flow data, only pressure data, combined flow and pressure data. It was shown that for estimating arterial wall properties, flow measurements contain a larger amount of information about the parameters.

Müller applied UKF for the 1D blood flow model. The data was in frequency domain since clinical measurements are often related to max/min values, average values and frequency spectra retrieved throughout the cardiac cycle. The time step in UKF was equal to the duration of the cardiac cycle. It was shown that the filtering in frequency domain converges faster than filtering in time domain in the case of estimating Young's modulus [30].

The arterial stiffness can be obtained from Pulse Wave Velocity (PWV), Lombardi compared this method against Unscented Kalman Filtering and proved that UKF is more precise. The drawback of UKF is the computational cost

and data amount need to perform the filtering [31].

Bertoglio [33] identified the stiffness distribution of a silicon rubber aortic phantom using the FSI model. The material was mechanically tested by filling the tube with water and relating the exerted pressure by the fluid to the wall displacement using a linear elastic model, which is a function of Young's modulus, Poisson's ratio, tube thickness, and tube length. The estimated Young's modulus was reported to agree with the values obtained by mechanical testing. Additionally, the stiffness was higher in regions close to the ends of the tube where it was fixed to non-deforming material, therefore limiting the wall's radial displacement. The estimation procedure was then tested on clinical data from the aorta. It was assumed that the wall behaves according to the Mooney-Rivlin model. It was reported that for clinical data there were issues with convergence of the parameters to be estimated. The oscillations according to the author compensate for modelling and data errors.

Chapter 2

Theory

2.1 Continuum mechanics

Continuum mechanics is a subject unifying solid mechanics, fluid mechanics, thermodynamics and heat transfer. The matter is considered as a continuum, and we are interested in the volume-averaged behavior rather than the behavior at a molecular level. The continuum assumption tends to be reasonable when $\delta/\lambda << 1$ where δ is a characteristic length scale of the structure and λ is a characteristic length scale of the problem of interest [43]. This section focuses on an introduction to continuum mechanics of solid materials with a main interest on linear elastic and hyperelastic materials. We will start with displacement and strain definitions, introducing deformation and displacement gradient, then follow to various stress definitions, hence building the background for defining constitutive laws.

2.1.1 Displacement and strain

In continuum mechanics a particle is defined as a portion of the continuum body being considered. A particle has a prescribed position in space by a vector of origin O of the coordinate system $\mathbf{r} = [x_1, x_2, x_3] = x_i$. By convention, lowercase $\mathbf{x}$ refers to the deformed configuration and capital $\mathbf{X}$ to the reference configuration, which is typically the undeformed state. Thus, a displacement may be defined by:

$$u(\mathbf{X}, \mathbf{t}) = \mathbf{x}(\mathbf{X}, t) - \mathbf{X} \tag{2.1}$$

Considering an infinitesimal element dX we might be interested in the mapping of the reference configuration to its new positions dx. The deformation gradient links these two positions via:

$$dx = \mathbf{F}dX \tag{2.2}$$

where F is:

$$\mathbf{F} = \frac{\partial x}{\partial X} \tag{2.3}$$

or in tensor notation:

$$F_{ij} = \frac{\partial x_i}{\partial X_j} \tag{2.4}$$

Substituting x with $u+X$ from Eq.2.1 we arrive with the Displacement Gradient Tensor $\mathbf{H}$ also written as $\nabla\mathbf{u}$:

$$F_{ij} = \frac{\partial x_i}{\partial X_j} = \frac{\partial u_i + X_i}{\partial X_j} = \frac{\partial u_i}{\partial X_j} + 1 \tag{2.5}$$

we arrive with the Displacement Gradient Tensor $\mathbf{H}$ also written as $\nabla\mathbf{u}$ which is the last derivative [43, 44]:

$$\mathbf{H} = \nabla\mathbf{u} = \frac{\partial u_i}{\partial X_j} \tag{2.6}$$

Longitudinal strain ϵ in the direction $\mathbf{e}$ is defined as a ratio of the change in length ds of the material to the original length ds_0:

$$\epsilon = \lim_{s\to 0} \frac{s - s_0}{ds_0} = \frac{ds - ds_0}{ds_0} = \frac{ds}{ds_0} - 1 \tag{2.7}$$

Where s is the length of the material after applying force in the direction $\mathbf{e}$. The derivative $\frac{ds}{ds_0}$ is called stretch λ and it is the ratio between the length of the deformed material to to length of the undeformed material. Therefore the equation above can be written in terms of stretch:

$$\epsilon = \lambda - 1 \tag{2.8}$$

The Lagrange strain tensor is defined as:

$$\mathbf{E} = \frac{1}{2}\left(\mathbf{F}\mathbf{F}^T - \mathbf{I}\right) = \frac{1}{2}\left(\mathbf{H} + \mathbf{H}^T + \mathbf{H}^T\mathbf{H}\right) \tag{2.9}$$

Substituting the definition of the displacement gradient tensor $\mathbf{H}$ to the definition of Lagrange (Green) strain tensor, it may be rewritten in a derivative form:

$$\mathbf{E}_{kl} = \frac{1}{2}\left(\frac{\partial u_k}{\partial \mathbf{X}_l} + \frac{\partial u_l}{\partial \mathbf{X}_k} + \frac{\partial u_i}{\partial \mathbf{X}_k}\frac{\partial u_i}{\partial \mathbf{X}_l}\right) \tag{2.10}$$

which will simplify in the case of small deformations to a linear relationship (section 2.3).

2.1.2 Stress

The stress or traction vector t is defined as the ratio of force F to the area A:

$$\mathbf{t} = \lim_{dA \to 0} \frac{\mathbf{dF}}{dA} \tag{2.11}$$

When the area is on in the deformed configuration, we consider the Cauchy Stress (true stress). When the area is in the reference configuration (before the deformation occurred) it is called the engineering stress or nominal stress. Considering an infinitesimal cube with its surfaces perpendicular to the coordinate axes, the base vectors e_k being normal to cube's surfaces. The components of the stress vectors are denoted T_{ik} [44]:

$$t_k = T_{ik} e_i \tag{2.12}$$

The components of the stress vector $\mathbf{t}$ in the direction $\mathbf{n}$ (normal vector to the surface on which the force acts) and in the tangent plane to the surface A are the normal stress σ and the shear stress τ respectively. Cauchy's stress theorem states that the stress vector t on a surface through a particle P is uniquely determined by the stress tensor T in the particle and the unit normal $\mathbf{n}$ to the surface [44]:

$$\mathbf{t} = \mathbf{T}\mathbf{n} \tag{2.13}$$

Where $\mathbf{T}$ is the Cauchy Stress Tensor. Alternative stress measures to the Cauchy Stress Tensor are Piola - Kirchoff, Biot, Green, Naghdi or Mandel [45]. First Piola-Kirchoff stress tensor $\mathbf{P}$ is defined in terms of deformation gradient $\mathbf{F}$ and Cauchy's Stress tensor $\mathbf{T}$:

$$\mathbf{P} = J\mathbf{T}\mathbf{F}^{-T} \tag{2.14}$$

where $J = det(\mathbf{F})$. The first Piola-Kirchoff stress tensor can be interpreted as the force in the deformed configuration per unit area of the undeformed configuration. Second Piola-Kirchoff stress tensor S is defined by the relation:

$$\mathbf{S} = \mathbf{F}^{-1}\mathbf{P} = J\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T} \tag{2.15}$$

and can be viewed as the force in the undeformed configuration per unit area of the undeformed configuration.

2.2 Hyperelastic materials

Hyperelastic materials are materials that have the ability to store the strain imposed on the material in the form of energy. The stored energy $\psi(\mathbf{F})$ is not dissipated to the environment when the material returns to its undeformed state. The hyperelastic material is defined in terms of the strain energy density function ψ which can be expressed in terms of the deformation gradient $\psi(\mathbf{F})$ or the Cauchy stress tensor $\psi(\mathbf{C})$. The strain energy function is required to be zero when the material is in the reference configuration ($\mathbf{F} = \mathbf{I}$):

$$\psi(\mathbf{F} = \mathbf{I}) = 0 \tag{2.16}$$

Additionally, if the body is stretched infinitely ($det(F) \to \infty$ or compressed to a point ($det(F) \to 0$, the strain energy is required to tend to infinity [45]:

$$\psi(\mathbf{F}) \to \infty \tag{2.17}$$

From equation 2.15 it follows that the Cauchy stress tensor may be rewritten in terms of the second Piola-Kirchoff stress tensor as:

$$\mathbf{T} = J^{-1}\mathbf{F}\mathbf{S}\mathbf{F}^T \tag{2.18}$$

given that the second Piola-Kirchoff stress tensor has a relationship with strain energy function [45, 46]:

$$\mathbf{S} = 2\frac{\partial\psi}{\partial\mathbf{C}} \tag{2.19}$$

where $\mathbf{C}$ is the right Cauchy-Green tensor defined as:

$$\mathbf{C} = \mathbf{F}^T\mathbf{F} \tag{2.20}$$

whereas left Cauchy stress tensor is:

$$\mathbf{b} = \mathbf{F}\mathbf{F}^T \tag{2.21}$$

Therefore the true (Cauchy) stress may be related to the strain and definition of hyperelastic material in terms of strain energy function by combining equations 2.18 and 2.19. For isotropic hyperelastic material the strain energy function ψ does not depend on the direction of applied force therefore it is written in terms of the invariants of $\mathbf{C}$ [46]:

$$\mathbf{T} = 2J^{-1}\psi_1 b + 4J^{-1}\psi_2 b^2 + 2J\psi_3\mathbf{I} \tag{2.22}$$

Where ψ_1, ψ_2 and ψ_3 are the derivatives of ψ with respect to the invariants of $\mathbf{b}$; the fist (I_1), second (I_2) and third (I_3) invariants of $\mathbf{b}$ respectively.

- Incompressible material (det(F)=J=1):

$$I_1 = tr(b) = \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \tag{2.23}$$

$$I_2 = \lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_2^2\lambda_3^2 \tag{2.24}$$

$$I_3 = \lambda_1^2\lambda_2^2\lambda_3^2 = 1 \tag{2.25}$$

- Compressible material (det(F) $\neq$ 1):

$$I_1 = J^{2/3}\left(\lambda_1^2 + \lambda_2^2 + \lambda_3^2\right) \tag{2.26}$$

$$I_2 = J^{4/3}\left(\lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_2^2\lambda_3^2\right) \tag{2.27}$$

$$I_3 = \lambda_1^2\lambda_2^2\lambda_3^2 \neq 1 \tag{2.28}$$

2.2.1 Neo-Hookean Material

For compressible Neo-Hookean material, four different strain energy density functions were found [47, 46]:

$$\Psi = \frac{\mu}{2}(I_1 - 3) - \mu lnJ + \frac{\lambda}{2}(lnJ)^2 \tag{2.29}$$

$$\Psi = \frac{\mu}{2}(I_1 - 3) + a_1(J-1)^2 + a_2 lnJ \tag{2.30}$$

$$\Psi = \frac{\mu}{2}(I_1 J^{b_1} - 3) + b_2\left(J^2 + \frac{1}{J^2} - 2\right) \tag{2.31}$$

$$\Psi = \frac{\mu}{2}(I_1 - 3) + c_1(J^{c_2} - 1) \tag{2.32}$$

where a_1, a_2, b_1, b_2, c_1 and c_2 are adjustable constants. The parameters μ and λ are functions of Young's Modulus E and Poisson's ratio ν:

$$\mu = \frac{E}{2(1+\nu)} \tag{2.33}$$

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \tag{2.34}$$

The Eq. 2.29 is the form of the strain energy density equation implemented in the open-source software FEBio [48]. It is unknown to the author of this thesis why such a definition of compressible Neo-Hookean material was chosen by the developers of the software. The implementation was tested by the authors against analytical solution for the confined compression creep test and unconfined compression stress relaxation test. For both tests a cylindrical sample was used. It is claimed by the authors and shown that the agreement of simulated results are in agreement with the analytical solution [48, 49].

2.3 Isotropic Linear Elastic Material

For small deformations and small displacements, the Green strain tensor (Eq. 2.10 simplifies to:

$$\mathbf{E}_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial \mathbf{X}_j} + \frac{\partial u_j}{\mathbf{X}_j}\right) \tag{2.35}$$

The model is suitable if all of the following constraints are fulfilled [44]:

- Longitudinal strain $\ll 0.001$
- Shear strain $\ll 0.02$
- Absolute value of the volumetric strain is less than 0.03

In FEBio the isotropic linear elastic model is given in terms of strain energy function:

$$\Psi = \frac{1}{2}\lambda(tr\mathbf{E})^2 + \mu\mathbf{E} : \mathbf{E} \tag{2.36}$$

2.4 Inverse problems

An inverse problem is a type of problem that relates to the determination of unknown causes or inputs based on observation of their effects. That is the opposite of the direct problem in which the effects or outputs are determined based on a complete description of the inputs [50]. The inverse problem may focus on reconstructing the boundary conditions, unknown parameters of the governing model, reconstructing initial conditions or even reconstructing the geometry [51, 52, 53]. In general, the direct problem is considered a well-posed problem. The problem may be formulated as a relationship between some property x to be estimated and the measurement z. The forward operator A is a map that maps x to z.

$$Ax = z \tag{2.37}$$

For the problem to be well posed it must fulfill all of the following conditions [54, 55, 56]:

- For every measurement vector $z \in Z$ there exists a parameter solution x
- The solution must be unique

- The solution must be stable with respect to perturbation on the right-hand side, that is the operator A^{-1} must be defined throughout the space Z and be continuous

If any of the above conditions is violated the well-posed problem becomes ill-posed problem and is unsolvable. Some techniques available to solve the inverse problem are maximum likelihood estimation (MLE) and maximum a posteriori (MAP).

2.4.1 Maximum Likelihood Estimation

The MLE estimates the unknown parameters from the following perspective. Given the measurement, what are the parameters of the model from which these observations would most likely produce the data. To solve this problem, a likelihood function L is introduced:

$$L(x|z) = f(z|x) \tag{2.38}$$

Thus $L(x|z)$ represents the likelihood of the parameter x given measurement z. Where f is the probability density function which specifies the probability of observing data vector z given the parameter x [57, 51]. If the inverse problem is linear and the data errors are independent and normally distributed with known standard deviation σ and mean equal to zero, then the MLE reduces to least squares. The probability density function (PDF) for i-th measurement z_i takes the form of:

$$f_i(z_i|x) = \frac{1}{\sigma_i\sqrt{2\pi}} exp(-0.5(z_i - (Ax)_i^2)/\sigma_i^2 \tag{2.39}$$

Given the assumption of independent measurements, the above equation can be expressed as multiplication of PDFs for the whole data set:

$$f(z = (z_1, z_2, ..., z_n|x) = f_1(z_1|x)f_2(z_2|x) \cdots f_n(z_n|x) \tag{2.40}$$

Taking the natural logarithm and plugging back in the normal distribution function from Eq. 2.39 to Eq. 2.40, the expression above is then maximized to find the the most likely parameter x:

$$max\ log \prod -0.5\frac{z_i - (Ax)_i^2}{\sigma_i^2} = max \left[-0.5 \sum \frac{z_i - (A\theta)_i^2}{\sigma_i^2}\right] \tag{2.41}$$

Changing the sign, the problem is turned into minimization problem and the constant 0.5 is often dropped:

$$\hat{x}_{MLE} = min \sum \frac{z_i - (Ax)_i^2}{\sigma_i^2} \tag{2.42}$$

2.4.2 Maximum a posteriori

If prior information on the parameters to be estimated is available, the information can be used in the Maximum a posteriori (MAP) framework. The MAP is based on Bayes' theorem and is therefore called a Bayesian estimator. With Bayes's theorem, Eq.2.43, it is possible to calculate the posterior distribution of the parameter x given new information (measurement z).

$$p(x|z) = \frac{p(z|x) \cdot p(x)}{\int_{-\infty}^{\infty} p(z|x) \cdot p(x) dx} \tag{2.43}$$

The denominator in the Eq. 2.43 normalizes the posterior distribution so that it sums up to unity. The above equation is often written as a proportionality:

$$p(x|z) \propto p(z|x) \cdot p(x) \tag{2.44}$$

Maximum a posteriori method maximizes the natural logarithm of posterior distribution in Eq.2.44 [51, 56]:

$$\hat{x}_{MAP} = arg\ max\ \left[logP(x) + \sum logP(z_n|x) \right] \tag{2.45}$$

2.4.3 Regularization

Small perturbations (errors) in the experimental data in Eq.2.37 often lead to an unstable and useless solution of the inverse problem. Naive solution, by solving the Eq. 2.37 by multiplication of both sides by the inverse of matrix A ($x = A^{-1}z$) or the least squares solution leads to the wrong solution far from the expected solution.

Regularization is therefore needed to solve the issue of instability, non-uniqueness of the solution, and results far from the exact solution [58]. A measure of the instability of the solution is the condition number, which is defined as a ratio of the maximum and minimum values of the singular values s (eigenvalues) of the matrix A. The eigenvalues are obtained from the singular value decomposition (SVD) of matrix A [59], which will be discussed briefly in a moment. The definition of condition number of matrix A is as follows:

$$cond(A) = \frac{max(s)}{min(s)} \tag{2.46}$$

The SVD is an approximation of the matrix and not all components of the decomposition are needed for good approximation. Therefore by truncating

the SVD it is possible to cut off the components dominated by the noise (components with smallest eigenvalues) while keeping a good approximation. This approach has a benefit of lowering the condition number by increasing the value in the denominator in Eq. 2.46 therefore increasing the stability of inverse problem [51, 58, 60]. It shouldn't be a surprise that SVD is for this reason used in inverse problems. The shortcoming of the SVD is its computational time which may be prohibitive for large-scale problems.

Other method which is often used for regularization and does not rely on eigenvalues is Tikhonov regularization [51]. Both methods will be briefly discussed in the following sections.

Singular Value Decomposition

Singular Value Decomposition factors matrix $A \in R^{mxn}$ with $m \geq n$ into three matrices:

$$A = USV^T = \sum_{i=1}^{n} u_i s_i v_i^T \tag{2.47}$$

Where

- $U \in R^{mxn}$ is a matrix with orthonormal columns:

$$u_i^T u_j = \delta_{ij} \tag{2.48}$$

  where δ_{ij} is the Kronecker's delta

- $S \in R^{nxn}$ is the diagonal matrix of non-negative singular values with components s satisfying:

$$s_1 \geq s_2 \geq ... \geq s_n \geq 0 \tag{2.49}$$

- $V \in R^{nxn}$ is a matrix with orthonormal columns satisfying equation 2.48

Performing SVD on matrix A and the vector of measurements z it is possible to solve the system of equations $Ax = z$ by:

$$x = A^{-1}z = \sum_{i=1}^{n} \frac{u_i^T z}{s_i} v_i \tag{2.50}$$

When truncating the Singular Value Decomposition a truncation parameter k is chosen such that the noise-dominated SVD coefficients are discarded. The

condition number new matrix A_k which is an approximation of original matrix A then reduces since $s_k > s_n$ [58, 59]:

$$cond(A_k) = \frac{s_1}{s_k} \tag{2.51}$$

Tikhonov regularization

As mentioned above, the SVD approach may not be the best technique for large-scale problems due to the computational time required to perform the decomposition. An alternative method is the Tikhonov regularization, where the regularization parameter λ is introduced into the minimization as follows:

$$\min_x \{||Ax - z||_2^2 + \lambda^2 ||x||_2^2\} \tag{2.52}$$

where $||Ax - z||_2^2$ is the squared Euclidean norm of the residuals. Parameter λ gives preference to solutions with smaller norms. If the parameter is set to zero, then the problem becomes ordinary least squares, a non-zero value is known as a ridge regression. Changing the value of λ influences the condition number of matrix A therefore regularizing the problem [61]. In order to choose the proper value of λ the minimization problem is solved for a range of values. The log-log plot of the solution norm $||x||_2^2$ versus residual norm $||Ax - z||_2^2$ forms an L shaped curve referred to as L-curve [62]. The value for which the solution is the closest to the corner of the L-curve is then selected [51].

2.5 Kalman filtering

Most of the difficulties in inverse problem arise from the ill-conditioned problem. Regularization techniques are a solution to overcome the problem of instability of the solution.

Bayesian inference tackles the problem from a different perspective. Rather than looking for a single value of the parameter of interest, the solution is given as a probability distribution [63]. The solution obtained based on Bayesian approach may produce equivalent results to the least squares and regularized problems [51, 64].

Kalman Filter is a special case of Recursive Bayesian Estimation, under the assumption of a linear model and normally distributed random variables there exists a closed-form solution to the Bayes filter [65]. Kalman filter, also called

the linear quadratic estimator (LQE) because it minimizes the mean-squared estimation error of a system using noisy sensor measurements [66, 67]. It predicts future state of a system based on previous and current states in two steps which are repeated alternatively, prediction and update steps. In the prediction step, the recursive mathematical model of the system is used to predict the state in the next time step.

The predicted value is called a prior. In the update step, the prediction is compared with the measurement by calculating the residual. The prediction is then corrected based on the residual, therefore updating belief about the system. The updated value is called posterior [68, 67].

The linear Kalman filter is suitable for processes governed by a linear model with Gaussian noise. In the case where the governing equations are non-linear, an extended Kalman filter (EKF) may be utilized to linearize the equations about the current mean and covariance [69]. The linearization is done by expanding the state function using Taylor series. That requires calculating partial derivatives in each time step, and the filter may become too computationally expensive.

If the model is too complicated, EKF may be not converge and other variations of the filter are used. The unscented Kalman filter (UKF) is often a weapon of choice for dealing with cases when EKF fails. UKF performs the Unscented Transform for better estimation of mean and covariance passed through the nonlinear model. It is achieved by carefully picking points called sigma points from an arbitrary probability distribution and weights associated with each sigma point and passing these points to the nonlinear equations. Using the output of the nonlinear functions and chosen weights, the mean and covariance of the output can be retrieved [70]. The visualization of the working principles of Kalman filter is described in the appendix 8.1.

2.5.1 Linear Kalman Filter for state estimation

The Kalman filter estimates the state of a linear dynamic system. It utilises the process and measurement model with noisy input-output measurement signals.

The algorithm has two steps, prediction and update. In the prediction step, the Kalman Filter estimates the current state variables and their uncertainties. In the update step, these estimates are updated using a weighted average by taking into consideration the measurement of the real system. The mathematical

description of the process has a form of:

$$x_k^- = Ax_{k-1}^+ + Bu_{k-1} + w_{k-1} \tag{2.53}$$

where x is the state of the system, for example position of a vehicle. Matrix A relates previous time step $k-1$ to the state at the current step k. Matrix B relates optional control input u to the state x. The process noise is represented by w and it follows Gaussian distribution with zero mean and covariance Q:

$$w \sim N(0, Q)$$

The measurement z at time step k is related to the true state of the system x by matrix H:

$$z_k = Hx_k + v_k \tag{2.54}$$

The v vector is a measurement noise and it follows Gaussian normal distribution with zero mean and covariance R:

$$v \sim N(0, R)$$

The whole algorithm for estimating and predicting state of the system is presented below.

- Prediction step
    1. Predict current step based on previous state

$$x_k^- = Ax_{k-1}^+ + Bu_{k-1} \tag{2.55}$$

    2. Estimate covariance

$$P_k^- = AP_{k-1}^+ A^T + Q \tag{2.56}$$

- Measurement update
    1. Calculate Kalman gain

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \tag{2.57}$$

    2. Update state estimate

$$x_k^+ = x_k^- + K_k(z_k - Hx_k^-) \tag{2.58}$$

3. Update covariance

$$P_k^+ = (I - K_k H) P_k^- \tag{2.59}$$

where I is the identity matrix

At first time step, the predicted state $\hat{x}_k^-$ and covariance matrix P_k^- have to be initialized by guessing their values. As soon as the measurements are available the update step can be performed. Then the algorithm runs recursively from equation Eq.2.55 to Eq.2.59 [71, 69].

2.5.2 Extended Kalman Filter state estimation

The Extended Kalman Filter (EKF) linearizes the nonlinear mapping function f from previous time step $k-1$ to current time step k by utilizing Taylor series expansion. The mapping of state system to current step has the form:

$$x_{k+1} = f(x_k) + w_k \tag{2.60}$$

The measurement function h may be also nonlinear:

$$z_{k+1} = h(x_{k+1}) + v_{k+1} \tag{2.61}$$

The algorithm for EKF consists of following steps.

1. Initialize for $k = 0$

$$x_0 = E[x_0] \tag{2.62}$$

$$P_0 = E[(x_0 - E[x_0])(x_0 - E[x_0])^T] \tag{2.63}$$

EKF Algorithm for $k = 1, 2, 3, ..$

2. State estimation

$$x_k^- = f(k-1, x_{k-1}^+) \tag{2.64}$$

3. Covariance estimation

$$P_k^- = A_{k,k-1} P_{k-1}^+ A_{k,k-1}^T + Q_{k-1} \tag{2.65}$$

where F is the linearization of the model

$$A_{k|k-1} = \left. \frac{\partial f(,x)}{\partial x} \right|_{x=x_{k-1}^+} \tag{2.66}$$

4. Kalman Gain Matrix

$$K_k = P_k^- H^T (H P_k^- H^T + R_k)^{-1} \tag{2.67}$$

where H is the linearization of observation

$$H_{k|k-1} = \left. \frac{\partial h(x)}{\partial x} \right|_{x=x_k^-} \tag{2.68}$$

5. State estimation update

$$x_k^+ = x_k^- + K[z_k - h(k, x_k^-)] \tag{2.69}$$

6. Covariance estimation update

$$P_k^+ = P_k^- [I - K_k H_k] \tag{2.70}$$

The EKF suffers from sever limitations. First order linearization of the nonlinear system can be applied only if the Jacobian matrix exists. The linearization is reliable only if the error propagation can be well approximated by a linear function. If that condition is not met, it can produce large errors in the estimates and divergence of the filter [70].

2.6 Parameter Estimation

2.6.1 Extended Kalman Filter for parameter estimation

The general nonlinear form of recursive state transition function f_{k-1} and nonlinear measurement function h_k have the form:

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, \theta, w_{k-1}) \tag{2.71}$$

It is a function of the state at previous time step x_{k-1}, input vector at previous time step u_{k-1} and process noise w_{k-1} The measurement function:

$$z_k = h_k(x_k, u_k, \theta, v_k) \tag{2.72}$$

Where v_k represents the unknown true error assumed to be Gaussian noise with 0 mean.

Parameter θ is assumed to be time invariant, therefore it's evolution is modelled as:

$$\theta_{k+1} = \theta_k + r_k \tag{2.73}$$

The error r_k allows the filter to vary the unknown constant parameter.
The state of the system is modelled by function h_k associated with error e_k modeling the sensor noise and model error, therefore the measurement estimate is calculated by:

$$z_k^- = h_k(x_k, u_k, \theta_k, e_k) \tag{2.74}$$

The measurement function has to be differentiated with respect to the parameter θ, denoted as $\hat{C}_k^\theta$. It has to be noted that the state transition function x_k is a function of θ, therefore it requires calculating a total derivative by the chain rule.

$$C_k^\theta = \frac{dh_k(x_k, u_k, \theta, e_k)}{d\theta} = \frac{dh_k(x_k, u_k, \theta, e_k)}{d\theta} + \frac{dh_k(x_k, u_k, \theta, e_k)}{dx_k}\frac{dx_k}{d\theta} \tag{2.75}$$

$$\frac{dx_k}{d\theta} = \frac{\partial f_{k-1}(x_{k-1}, u_{k-1}, \theta, w_{k-1})}{\partial\theta} + \frac{\partial f_{k-1}(x_{k-1}, u_{k-1}, \theta, w_{k-1})}{\partial x_{k-1}}\frac{dx_{k-1}}{d\theta} \tag{2.76}$$

If the noise is not additive, then it is required to compute the derivative of the state transition function with respect to the modelled sensor error e_k, denoted as D_k^θ.

$$D_k^\theta = \frac{dh_k(x_k, u_k, \theta, e_k)}{de_k} \tag{2.77}$$

EKF algorithm. The step by step algorithm:

1. Initialization

$$\theta^+ = E[\theta_0] \tag{2.78}$$

$$P_{\theta,0}^+ = E[(\theta_0 - \theta_0^+)(\theta_0 + \theta_0^+)^T] \tag{2.79}$$

$$\frac{dx_0}{d\theta} = 0 \tag{2.80}$$

   The derivative $\frac{dx_0}{d\theta} = 0$ is initialized as 0 unless side information is available

2. Computation for k = 1, 2,... Parameter estimation time update

$$\hat{\theta_k}^- = \hat{\theta}_{k-1}^+ \tag{2.81}$$

Error covariance time update

$$P_{\theta,k}^{-} = P_{\theta,k-1}^{+} + P_{\tilde{r}} \tag{2.82}$$

Output estimate where e is the model error

$$z_k^{-} = h_k(x_k, u_k, \theta, \bar{e}_k) \tag{2.83}$$

Kalman gain matrix

$$K_k^{\theta} = P_{\theta,k}^{-}(C_k^{\theta})^T[(C_k^{\theta})P_{\theta,k}^{-}(C_k^{\theta})^T + (C_k^{\theta})P_{\bar{e}}^{-}C_k^{\theta})^T]^{-1} \tag{2.84}$$

Parameter estimation measurement update

$$\theta_{k-1}^{+} = \theta_{k-1}^{-} + K_k^{\theta}(z_k - \hat{z}_k) \tag{2.85}$$

Error covariance measurement update

$$P_{\theta,k}^{+} = (I - K_k^{\theta}C_k^{\theta})P_{\theta,k}^{-} \tag{2.86}$$

2.7 Simultaneous State and Parameter Estimation

2.7.1 Dual Extended Kalman Filter

The mathematical model of state dynamics explicitly includes the parameters as the vector θ:

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}, \theta) \tag{2.87}$$

$$z_k = h_k(x_k, u_k, v_k, \theta) \tag{2.88}$$

For estimation of a constant set of parameters θ, it's variation is again allowed by adding noise term r_{k-1}.

$$\theta_k = \theta_{k-1} + r_{k-1} \tag{2.89}$$

The measurement is estimated by embedding Eq. 2.87 in Eq. 2.88 giving

$$\hat{z}_k = h_k\left(f_{k-1}(x_{k-1}, u_{k-1}, \theta, w_{k-1}), u_k, e_k, \theta_{k-1}\right) \tag{2.90}$$

The Dual state and parameter estimation algorithm similarly to parameter estimation via EKF requires computing the total differential C_k^{θ}.

$$C_k^{\theta} = \frac{dh_k(x_k^{-}, u_k, \theta, e_k)}{d\theta} = \frac{dh_k(x_k^{-}, u_k, \theta, e_k)}{d\theta} + \frac{dh_k(x_k^{-}, u_k, \theta, e_k)}{dx_k^{-}}\frac{dx_k^{-}}{d\theta} \tag{2.91}$$

$$\frac{dx_k^-}{d\theta} = \frac{\partial f_{k-1}(x_{k-1}^+, u_{k-1}, \theta, w_{k-1})}{\partial \theta} + \frac{\partial f_{k-1}(x_{k-1}^+, u_{k-1}, \theta, w_{k-1})}{\partial x_{k-1}^+} \frac{dx_{k-1}^+}{d\theta} \quad (2.92)$$

$$\frac{dx_{k-1}^+}{d\theta} = \frac{dx_k^-}{d\theta} - L_{k-1}^x \frac{dh_{k-1}(x_{k-1}^-, u_{k-1}, \theta, e_k)}{d\theta} \quad (2.93)$$

Definitions:

$$A_k = \frac{df_k(x_k, u_k, w_k, \theta)}{dx_k} \quad (2.94)$$

$$A_k = \frac{df_k(x_k, u_k, w_k, \theta)}{dw_k} \quad (2.95)$$

$$C_k^x = \frac{dh_k(x_k, u_k, \theta_k^-, v_k)}{dx_k} \quad (2.96)$$

$$D_k^x = \frac{dh_k(x_k, u_k, \theta, v_k)}{dv_k} \quad (2.97)$$

$$C_k^\theta = \frac{dh_k(x_k^-, u_k, \theta, e_k)}{d\theta} \quad (2.98)$$

$$D_k^\theta = \frac{dh_k(x_k^-, u_k, \theta, e_k)}{de_k} \quad (2.99)$$

DEKF algorithm. The dual extended Kalman filter algorithm procedure:

1. Initialization for k = 0,

$$\theta_0^+ = E[\theta_0] \quad (2.100)$$

$$P_{\theta,0}^+ = E[(\theta_0 - \theta_0^+)(\theta_0 + \theta_0^+)^T] \quad (2.101)$$

$$x_0^+ = E[x_0] \quad (2.102)$$

$$P_{x,0}^+ = E[(x_0 - x_0^+)(x_0 + x_0^+)^T] \quad (2.103)$$

2. Computation for k=1,2,...
   Time update for the weight filter

$$\theta_k^- = \theta_{k-1}^+ \quad (2.104)$$

$$P_{\theta,k}^- = P_{\theta,k-1}^+ + P_{\tilde{r}} \quad (2.105)$$

   Time update for the state filter

$$x_k^- = f_{k-1}(x_{k-1}^+, u_{k-1}, \theta_k^-, \tilde{w}) \quad (2.106)$$

$$P_{x,k}^{-} = A_{k-1} P_{x,k-1}^{+} A_{k-1}^{T} + B_{k-1} P_{\tilde{w}} B_{k-1}^{T} \tag{2.107}$$

Measurement update for the state filter

$$L_k^x = P_{x,k}^{-} (C_k^x)^T [C_k^x P_{x,k}^{-} (C_k^x)^T + D_k^x P_{\tilde{v}} (D_k^x)^T]^{-1} \tag{2.108}$$

$$x_k^{+} = x_k^{-} + L_k^x [z_k - h_k(x_k^{-}, u_k, \theta_k^{-}, \bar{v}] \tag{2.109}$$

$$P_{x,k}^{+} = (I - L_k^x C_k^x) P_{x,k}^{-} \tag{2.110}$$

Measurement update for the weight filter

$$L_k^\theta = P_{\theta,k}^{-} (C_k^\theta)^T [C_k^\theta P_{\theta,k}^{-} (C_k^\theta)^T + D_k^\theta P_{\tilde{e}} (D_k^\theta)^T]^{-1} \tag{2.111}$$

$$\theta_k^{+} = \theta_k^{-} + L_k^\theta [z_k - h_k(x_k^{-}, u_k, \theta_k^{-}, \bar{e}] \tag{2.112}$$

$$P_{\theta,k}^{+} = (I - L_k^\theta C_k^\theta) P_{\theta,k}^{-} \tag{2.113}$$

Chapter 3

Experiments

The schematic representation of the experimental test setup is illustrated in 3.1.

The experimental rig was designed within ENTHRAL project to replicate the behavior of the left common carotid artery (LCCA) and allow direct monitoring of an elastic tube subjected to simulated cardiac pressure and flow cycles. To emulate physiological conditions in the LCCA of an adult male, the volumetric flow rate was maintained at approximately 0.5 l/min [72].

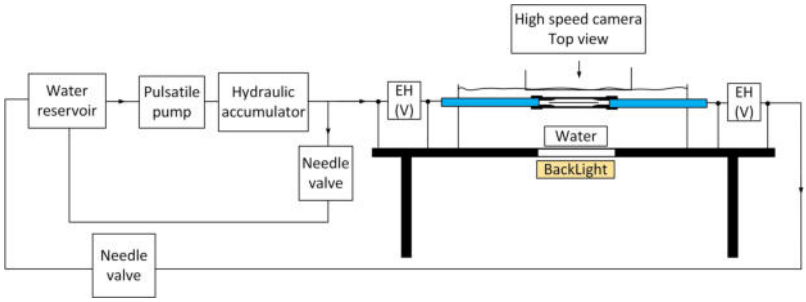
To maximize data collection within the physiological pressure range, measurements were conducted across four systole/diastole pressure ratios while maintaining 40 mmHg of pulse pressure: A) 110/70 mmHg, B) 120/80 mmHg, C) 135/95 mmHg, and D) 140/100 mmHg. These ratios were selected to cover a broad spectrum of physiological values and scenarios, categorized as: A - optimal, B - normal, C - high normal, and D - Grade 1 hypertension [73].

The flow was generated with Harvard Apparatus Pulsatile Blood Pump which mimics the ventricular action of the heart [74]. The specification allows to set stroke volume, stroke rate and systole/diastole ratio which were set to 15 ml, 60 r.p.m. and 35%/65% respectively.

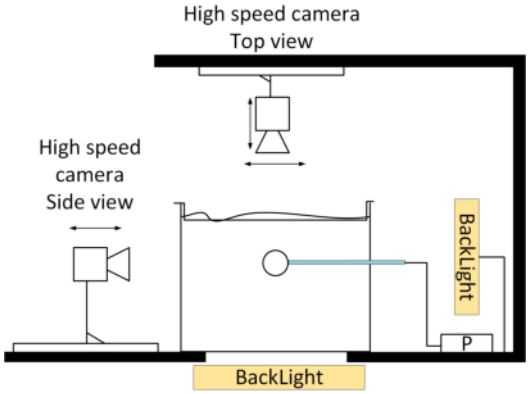
Water served as the working fluid and its flow rate was monitored with Endress+ Hauser Dosimag electromagnetic flow meter [75].

Eight Harvard Apparatus Blood Pressure Transducers (APT300) with Harvard Apparatus Compact Transducer Amplifiers were installed to measure pressure

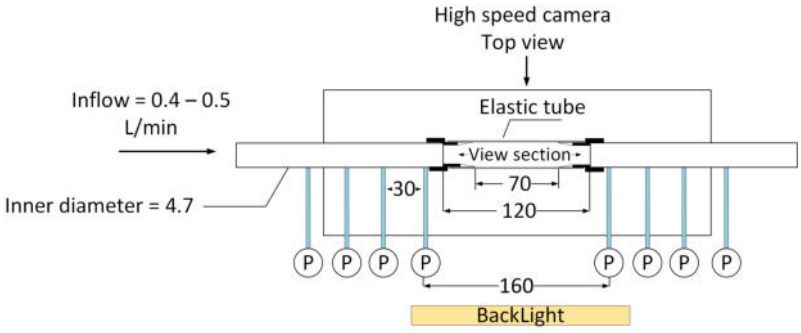
at eight points in total, four points proximal and four points distal to the phantom [76].



(a) Scheme of the entire testing rig.



(b) Side view of the testing rig.



(c) Detailed schematic of the distensible tube

Figure 3.1: a) Scheme of the entire testing rig. EH refers the Endress Hauser electromagnetic flow meters placed upstream and downstream of the phantom, b) side view of the testing rig and c) detailed schematic of the distensible tube with locations of the pressure gauges P. Units which are suppressed from the scheme are in millimeters. Adapted from [77].

The data collection was controlled by custom application programmed using

LabVIEW (National Instruments Corp., USA). The application enabled simultaneous acquisition of pressure and volumetric flow data from measurement devices. Pressure and flow data were collected every 10 ms, while images from camera were collected every 1 ms. Time synchronization of the data was achieved by field programmable gate array (FPGA) implemented in the cRIO 9074 controller.

The radial displacement of the tube was captured by two high-speed cameras: a Phantom MIRO C110 and a Phantom VEO 710 [78]. The Phantom MIRO C110 recording at 1000 frames per second (FPS) with a resolution of 1280x900, covered 22% of the entire tube. The camera was equipped with Irix lens [79] (150 mm f/2.8 Macro 1:1) together with a 20 mm extension tube. The Phantom VEO 710, equipped with a NIKKOR lens (200 mm F/4.0 MACRO) and an additional 68 mm extension tube, also recorded at 1000 FPS with a resolution of 1280x800, covering 30% of the tube.

The cameras were positioned perpendicularly to each other to capture tube deformations in two orthogonal directions (Fig. 3.2). Finally, two back lights were installed to eliminate background features and enhance the edges of the image from each camera.

For simultaneous measurement of displacement using ultrasound and camera, the Phantom VEO camera was replaced with ultrasound probe. A GE Healthcare ultrasound Vivid S70N machine with a ML6-15 linear probe was used [80]. The penetration depth was set to 1.5cm, and pixel resolution was 0.025 mm per pixel. The images were recorded over 30-45 s with an acquisition frequency of 59.9 FPS. For repeatability of the measurement the probe was fixed during recording using a clamp.

3.1 Camera image processing

At the beginning of each measurement, a calibration procedure was performed (Fig. 3.3, right upper corner). A linear scale suited for microscopy was placed in the same position that the phantom would occupy, and each camera recorded an image of this scale. The images were filtered with the Canny filter to detect edges. The Canny filter was applied with high and low thresholds of 100 and 200 to produce an image with pixel values of 0 everywhere except at the edges of the tube/linear scale. This procedure allowed to determine the mapping of each pixel to the length in mm. For MIRO camera the median scale was 112.51

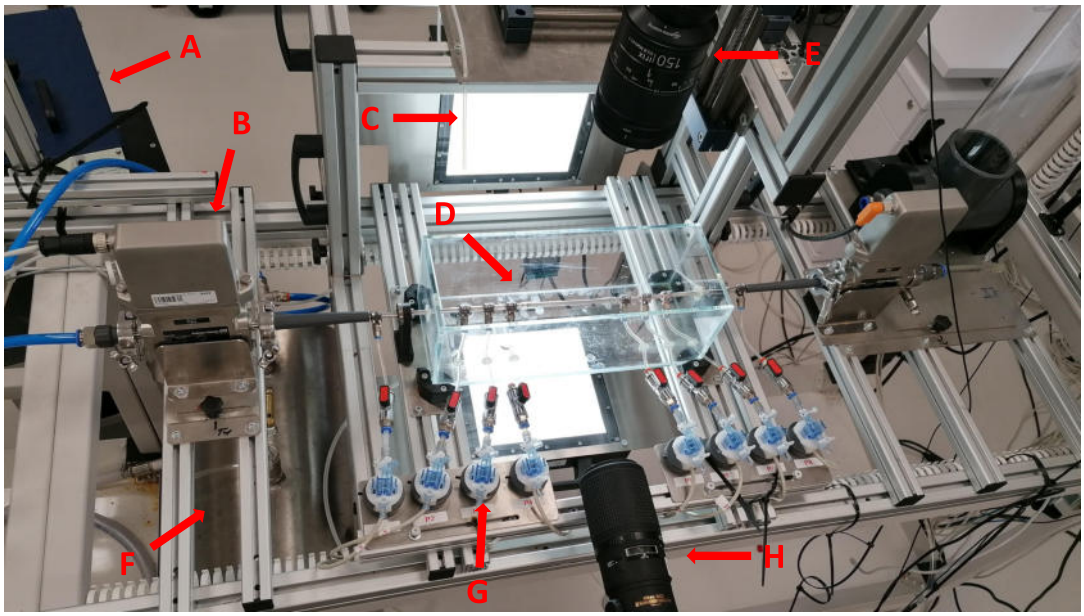


Figure 3.2: Configuration of the test rig with devices used for data acquisition with arrows indicating various elements of the set-up. A - pulsatile pump, B - flowmeter, C - back light, D - arterial phantom, E - top camera (MIRO), F - reservoir tank, G - pressure transducers, H - side camera (VEO). Adapted from [77].

pixels/mm while for VEO the median scale was 76.5 pixels/mm.

The lumen diameter was calculated by detecting the edges of the tube. Several points were chosen on the top and bottom edges. Then the distance in pixels between two opposite points was measured. The distance in pixels was then converted to mm which corresponded to the outer diameter of the tube. The average diameter at time t was calculated as the mean of all distances at time t. Finally, from the diameter change time series, the global minimum was subtracted (diameter at diastole) resulting in displacement time series. The workflow is represented graphically in the Fig. 3.3.

3.2 Ultrasound image processing

The ultrasound measurements and ultrasound measurements processing were carried out by dr inż. Jan Juszczyk (SUT, Department of Medical Informatics and Artificial Intelligence, Faculty of Biomedical Engineering).

The image processing of ultrasound measurements was carried out in six steps

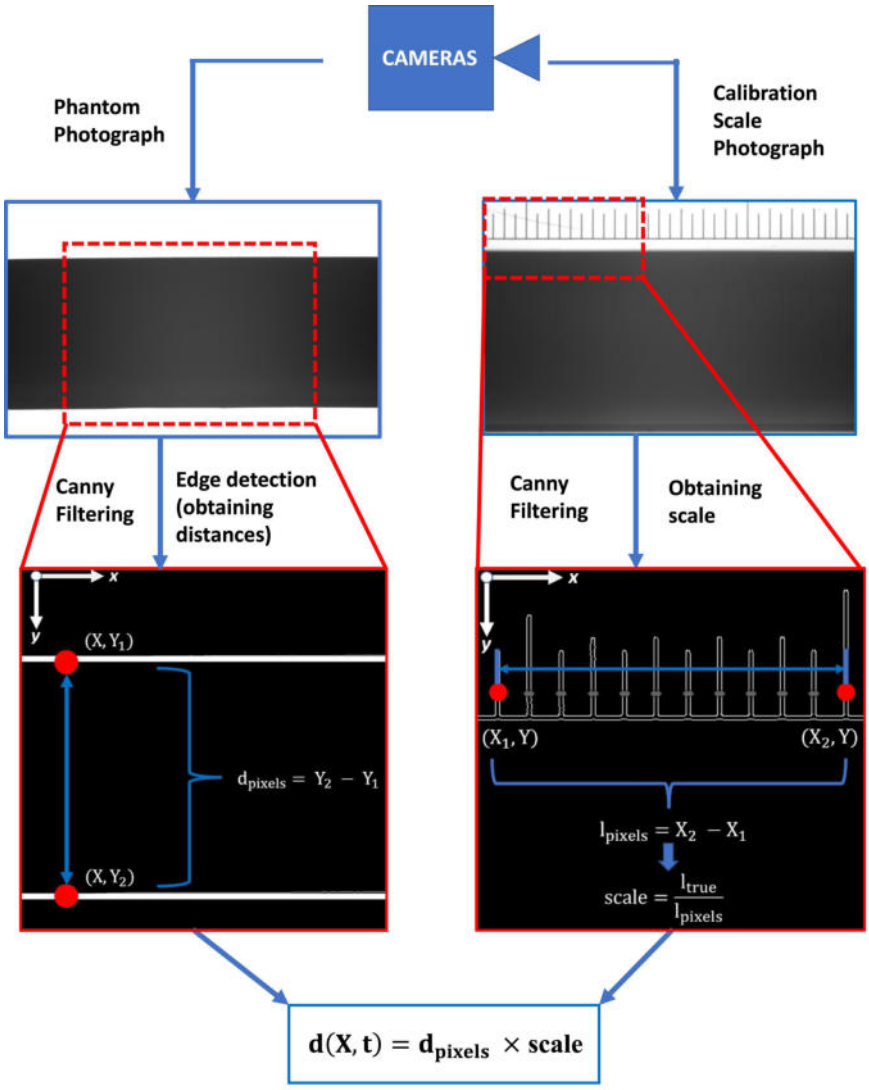


Figure 3.3: Workflow of the camera image processing. Adapted from [77].

and is shown in Fig.3.4 [77].

The grayscale values of the ultrasound image are first normalized (3.4a). The contrast between the wall and lumen of the phantom of the artery was then enhanced by morphological operation (3.4b). The color image was then reversed by the Otsu method (3.4c), resulting in a white background and black edges of the tube [81]. Lumen mask is applied to indicate the region occupied by the phantom lumen (3.4d). The edges of the vessel (3.4e) were identified using the Sobel filter [82]. The angle between the top and bottom edges was determined (3.4f) using the Hough transform [83]. The phantom's primary angle was de-

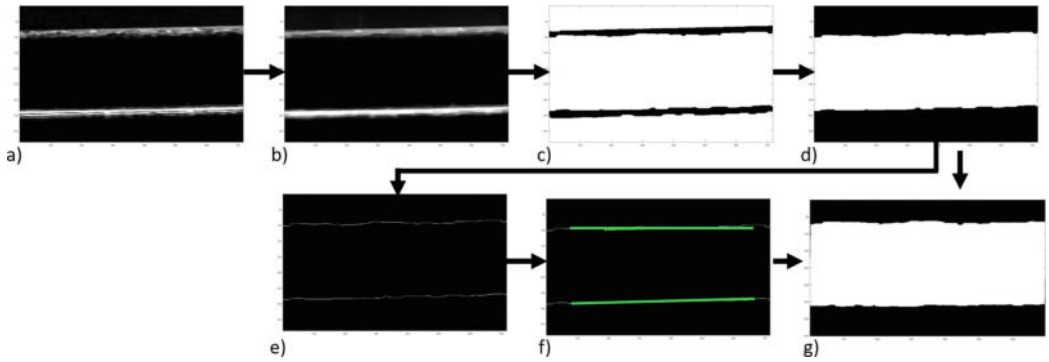


Figure 3.4: Ultrasound image processing, a) original image, b) grayscale adjustment, c) color inversion, d) lumen mask, e) internal wall edge detection, f) internal diameter and vessel angle detection, g) rotated lumen vessel mask. Adapted from [77].

fined as the average of the angles of the lower and upper edges.The lumen mask is rotated such that the primiary angle is parallel to the horizonal axis of the image (3.4g). The diameter was computed as the length of the cross-section perpendicular to the phantom's primary angle.

3.3 Medical data

The data were registered using a GE Healthcare ultrasound S70N machine with a ML6-15 linear probe [80]. Exemplary Ultrasound image for one of the patients is shown in the Fig.3.5. The edges visible in the figure are a result of image processing and represent the lumen of the artery. The ultrasound image processing is described in section 3.2.

The pressure waveform was measured with Atcor Sphygmocor CP\CVP\CPVH [84]. The data could not be exported to any file format other than .jpg (Fig 3.6) and a PlotDigitizer was used to read the data and was exported to a .csv file [85]. SciPy's interp1d function was used to generate data with a sampling time of 0.01s (Fig 3.7).

The ultrasound diameter measurement was processed in the same way to match the sampling rate of the pressure measurement. The sampling rate was chosen to match the sampling rate of registering the phantom's experimental data and to ensure the convergence of the estimation algorithm.

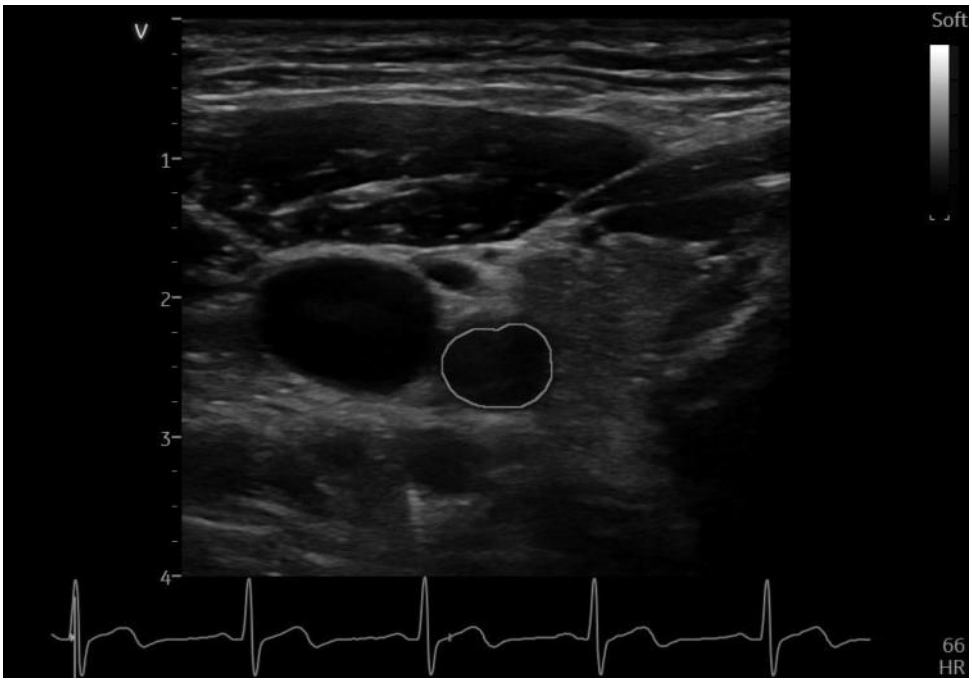


Figure 3.5: Process ultrasound image of one the patients. The visible edges represent the lumen of the artery.

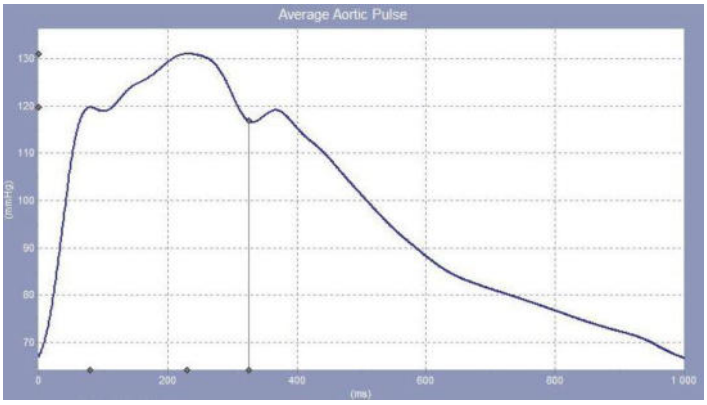


Figure 3.6: Pressure waveform for Patient3 captured in the image format by Atcor Atcor Sphygmocor CP/CVP/CPVH.

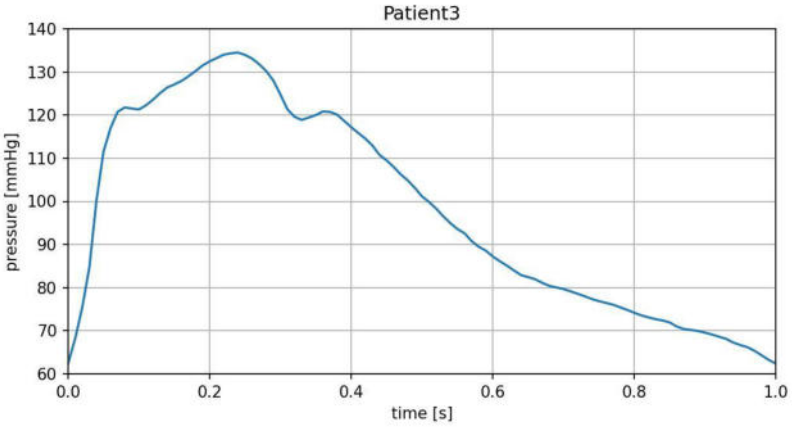


Figure 3.7: Pressure waveform for Patient3 processed by PlotDigitizer and Python.

Chapter 4

Proposed approach

Throught the entire thesis, FEBio software will be used as a Finite Element Method solver of the governing equations of solid mechanics. The choice of a constitutive model of the artery phantom is limited to the Linear Elastic and Neo-Hookean models.

Verification of the proposed approach to the inverse problem based on Kalman filtering is done using semi-experimental data. Experimental pressure waveforms were used to generate displacement data. The pressure waveforms registered during the experiments were used as a load on the internal wall of an idealized hollow cylinder with exactly known dimensions. Assuming a known value of Young's modulus and Poisson's ratio, the displacement of the outer wall of the cylinder was generated using analytical solution. The displacement data obtained from the analytical solution, was then fed to the Kalman filter, where the transition matrix was derived for the FEBio software (treated as a black box). Since the analytical solution is for linear elastic materials, the constitutive law of the same material was chosen in FEBio.

4.1 Mesh sensitivity and numerical model testing

4.1.1 Geometry, boundary conditions, material constants and mesh sensitivity

The geometry is assumed to be an idealized hollow cylinder with ends allowed to move in radial direction only. It is a fragment of a longer tube which was

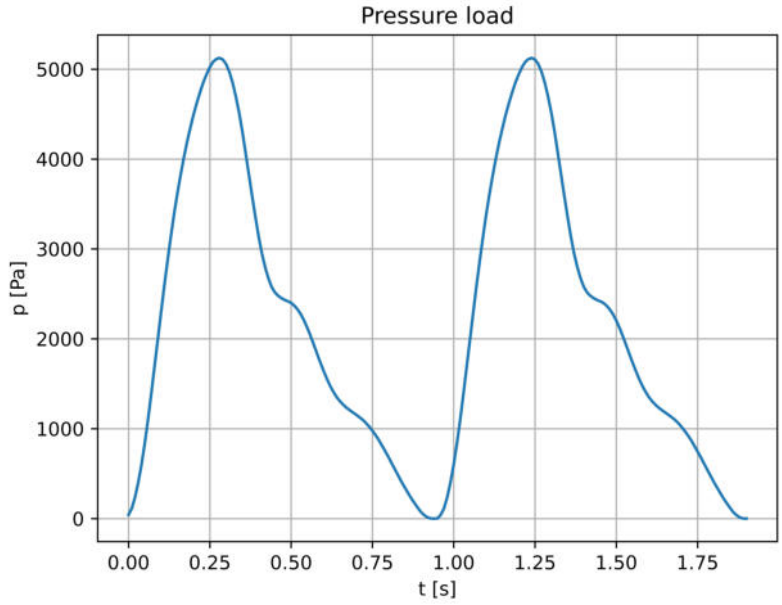


Figure 4.1: Pressure load for mesh sensitivity analysis (0-40 mmHg).

fixed on both ends. Its dimensions are:

- $r_{in} = 2.986$ mm
- $r_{out} = 3.33$ mm
- $L = 3$ mm

The cylinder's material is assumed to be linear elastic and the parameters of Young's Modulus E, Poisson's ratio ν and density of the solid material ρ_s are equal to:

- $E = 1.3$ MPa
- $\nu = 0.41$
- $\rho_s = 1000\ kg/m^3$

The pressure waveform was taken from experimental measurements and it was shifted by subtracting a global minimum. The resulting waveform is presented in Fig.4.1.

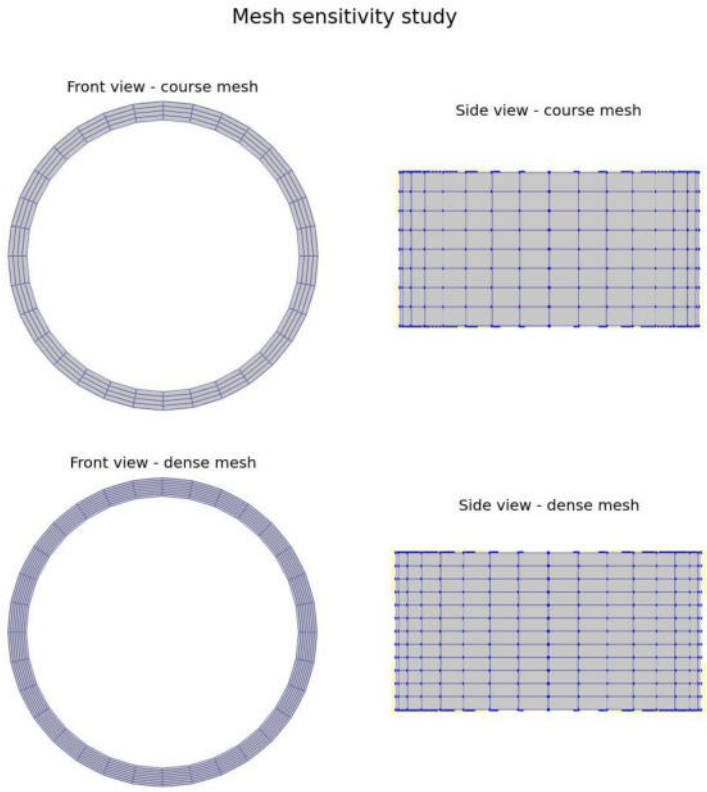


Figure 4.2: Front and side views of the investigated meshes for the sensitivity study.

4.1.2 Mesh sensitivity

Two finite element meshes were tested, the denser mesh counts around 2.5 times more nodes than the coarse mesh. The mesh was hexahedral with HEX8 elements, which means that the nodes were located at the vertices of the cube. Figure 4.2 shows the front and side views of the investigated meshes. Table 4.1 summarizes the number of elements and vertices count for each mesh.

Mesh	No. of elements	No. of nodes
Course	1024	1440
Dense	2880	3696

Table 4.1: Summary of the tested meshes.

These two meshes were used for both static and dynamic simulations. The resulting displacement waveforms were compared with each other using the formula given by the equation below[86]:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(z_i - \hat{z}_i)^2} \tag{4.1}$$

where n is the number of points and $\hat{z}_i$ is the estimate of measurement point z_i. The example of outer wall displacement waveforms obtained for static analysis of linear elastic material is presented in the Fig.4.3. The resulting errors for

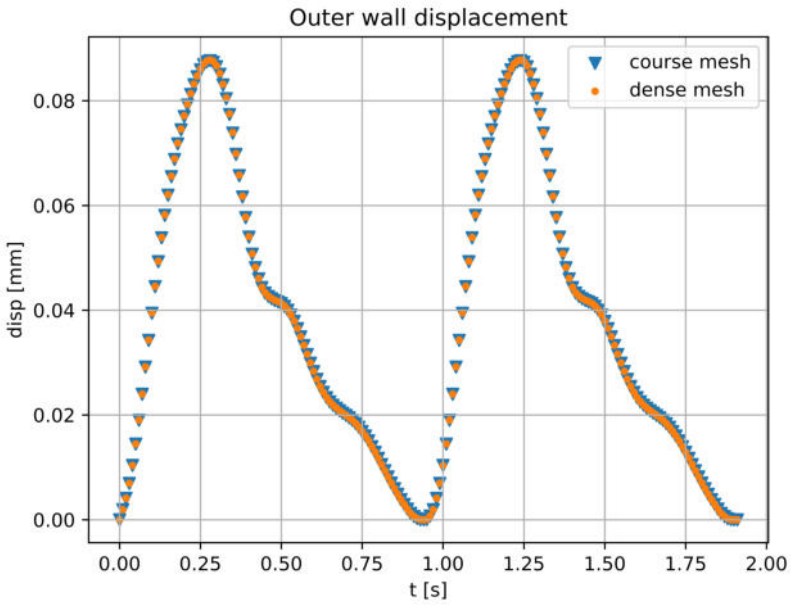


Figure 4.3: Outer wall displacement waveforms from static FEA (linear elastic model).

linear elastic and Neo-Hookean models for both static and dynamic simulations are summarized in the Table 4.2.

The errors change for the denser mesh is negligible, therefore the course mesh is used for further computations.

4.1.3 Static and dynamic finite element analysis

The model of the system dynamics is often written as a dynamic recursive equation, however, it is possible to use the Kalman filter for time-independent

Analysis	Model	RMSE
Static	Linear	0.000933
	Neo-Hookean	0.000936
Dynamic	Linear	0.000937
	Neo-Hookean	0.000938

Table 4.2: Summary of the tested meshes.

equations. One example would be finding the average resistance of a resistor given many noisy resistance measurements [87]. For this reason, a static and dynamic FEA was performed. If the results are the same then a static finite element model will be chosen to reduce the computational cost of the simulations. Since the course mesh is sufficient to achieve accurate results, the comparison between static and dynamic FEA is performed for the coarse mesh. The results are tabulated in Tab.4.3.

A visual proof that the curves overlap is shown in Fig.4.4 for linear elastic and Neo-Hookean models, respectively. The discrepancy of the results is negligible (around 0.006%) and the static FEA model is chosen for the inverse problem involving Kalman filtering.

Model	RMSE
Linear	0.005984
Neo-Hookean	0.006388

Table 4.3: Error between static and dynamic FEA.

4.1.4 Testing FEBio software for the Kalman filtering approach

It was shown in the previous section that static analysis can be used as a governing model for the linear elastic and Neo-Hookean artery phantom. In order to use that model successfully in the Kalman filtering procedure, it has to be checked whether stopping the simulation, changing the pressure load corresponding to the next time step and running the simulation again for the duration of one time step gives the same results as applying the whole time-varying load and not interrupting the process (e.g., Fig.4.4).

Using python programming language the test would look like:

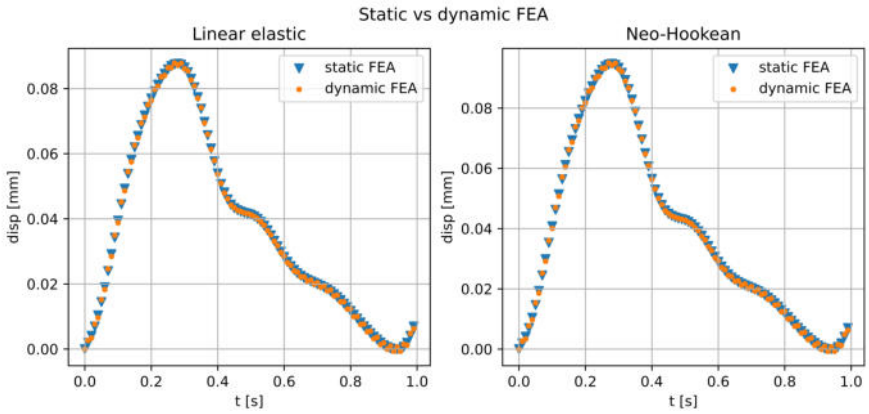


Figure 4.4: Static vs dynamic FEA results for linear elastic and Neo-Hookean model.

```
displacement = []
for p in pressures:
    displacement.append(run_febio(1.3, p))
```

where the displacement is an empty list. The for loop iterates over each pressure in a numpy array of experimental pressure values. The function run_febio takes two parameters, the first is Young's modulus [MPa], the second parameter is pressure [Pa] and returns displacement of some node on the outer wall of the cylinder. The function calls FEBio software to evaluate the displacement given a single value of pressure. The result of the function is added to the empty list with each iteration.

The results for linear elastic and the Neo-Hookean model are shown in the figure Fig.4.5. The fit between these two curves - test and "regular" FEA is much better for Neo-Hookean model. The overall fit for the linear model is 0.0002%. The overall fit of the Neo-Hookean model is 0.009%. At systolic pressure the relative displacement error is only 5e-5%. The worse fit of the Neo-Hookean can be explained by looking at the relative error at systolic pressure. The "test" displacement is higher than the "regular" FEA. The error is 0.9% at systole. This proves that the proposed approach can be used in further analysis.

4.2 Experimental data - phantom

The geometry dimensions of the phantom artery and the pressure under which they were tested are shown in Table 4.4

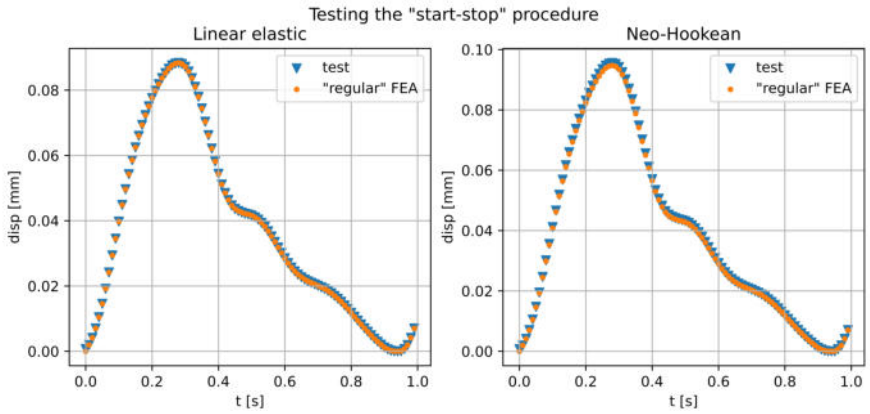


Figure 4.5: Results of the "start-stop" test.

Sample	outer r [mm]	h [mm]	h std [mm]	pressure [mmHg]
b3	2.986	0.347	0.00429	130/90
b4	2.913	0.338	0.00961	110/70
b5	2.871	0.355	0.011	120/80
b6	2.978	0.324	0.00445	140/100

Table 4.4: Artery's phantom geometrical data.

The sample here is treated as an artery phantom used during the experiment. After the experiment was finished, the phantom artery (sample) was tested for Young's modulus and Poisson's ratio. The Poisson's ratio is treated as a known parameter during for the inverse problem. The unknown quantity to be retrieved is the Young's modulus. The solution of the inverse problem is then compared to the experimental Young's modulus.

Mechanical properties of the samples were obtained with uniaxial tensile testing (dr hab. inż. Grzegorz Kokot, SUT, Department of Computational Mechanics and Engineering, Faculty of Mechanical Engineering) within ENTHRAL project. The measurements were carried out with the use of universal testing machine MTS Insight 10 kN [88]. The displacements were registered with the Q400-2D Dantec Dynamics system [89].

The average material thickness was determined using X-ray within ENTHRAL project (dr inż. Łukasz Krzemiński, SUT, Nanotechnology and Materials Technology Scientific and Didactic Laboratory, Faculty of Mechanical Engineering). The distribution of surface area vs. wall thickness is presented in Fig.4.6.

The displacement of the outer radius was recorded with the high-speed camera and the global minimum was assumed as the outer diastolic radius.

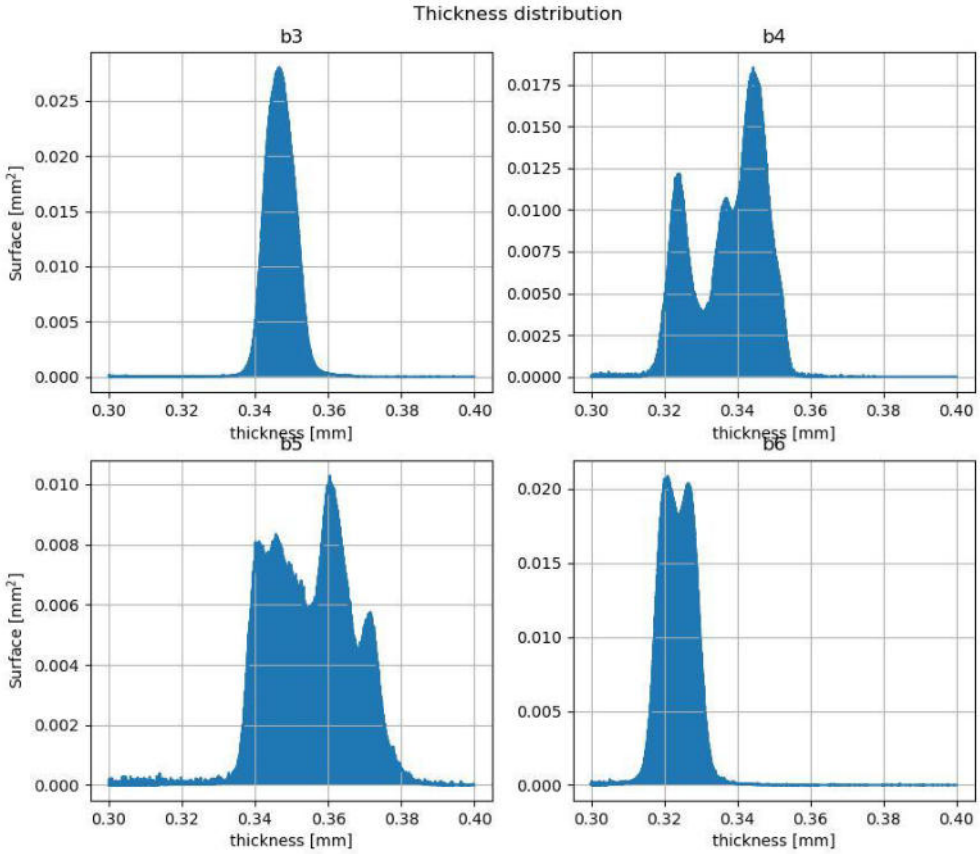


Figure 4.6: Samples' surface area vs wall thickness.

Since the the pressure and displacement curves for the analysis are averaged the mechanical properties from Tab.4.5 were also averaged and these averaged values were prescribed to the numerical model. The average values are listed in Table 4.6

Sample	E_{eng} [MPa]	E_{true} [MPa]	ν
b3_1	1.40	1.52	0.39
b3_2	1.47	1.62	0.41
b4_1	1.73	1.89	0.39
b4_2	1.60	1.75	0.43
b5_1	1.62	1.80	0.28
b5_2	1.39	1.53	0.31
b6_1	1.73	1.93	0.38
b6_2	1.95	2.16	0.48

Table 4.5: Mechanical properties of the samples from the end of the experiment. Each sample bX from experiment was cut in two halves bX_1 and bX_2.

Sample	$\bar{E}_{eng}$ [MPa]	$\bar{E}_{true}$ [MPa]	$\bar{\nu}$
b3	1.43	1.57	0.4
b4	1.66	1.82	0.41
b5	1.50	1.66	0.30
b6	1.84	2.05	0.43

Table 4.6: Average mechanical properties of the materials bX.

4.2.1 Patients data

The study was performed based on the consent of the bioethical committee (Resolution No 10/2021 by Bioethical Committee of the Silesian Chamber of Physicians, dated 01.03.2021). All patients were informed of the use of the gathered data and consented participation in the medical experiment within ENTHRAL project.

The patients' artery dimensions are summarized in table 4.7. For the medical data, Poisson's ratio is set to 0.499. The height, weight, age, sex and BMI are report in Tab. 4.8. Investigated patients report no medication use.

Patient	r_{in} [mm]	h [mm]
patient1	2.83	0.42
patient2	3.21	0.62
patient3	3.33	0.73

Table 4.7: Dimensions of the patients' LCCA.

Patient	Sex	Age	Height [cm]	Weight [kg]	BMI [kg/m]
Patient1	Female	32	153	54	23
Patient2	Male	28	186	94	27.17
Patient3	Male	51	180	70	21.6

Table 4.8: Additional data of each patient.

4.2.2 Experimental data processing - pressure and displacement

The length of each cycle in the experimental data is not constant. To fix this problem, the pressure and displacement data were processed the way shown in Figure 4.7.

In step 1 the maxima (systolic pressure/displacement corresponding to systolic pressure) were identified.

In step 2 the data was sliced into cycles using the indices (location in the numpy array) of each maximum value found in step 1. It is shown in the figure that at this step, the cycles vary in length.

In step 3 the cycle lengths were regularized so that each cycle has equal length. The data of each cycle were interpolated to find a function that matched the experimental data, then the function was used to generate data for the mean cycle length. If a cycle was shorter, then the data were extrapolated and the value corresponding to the end of the cycle was replaced with the value at its beginning for continuity.

At the end, in step 4, the results were averaged. The variance of the cycles for each sample is found in the table 4.9.

For the Kalman filtering approach, both raw and averaged data were tested. For the raw data it was difficult to achieve convergence of the Young's modulus estimate and averaged data was used for the inverse problem. Python code for the processing and creating plots such as the one in Fig.5.2 is found in Appendix 8.2.

Sample	var(p) [MPa^2]	var(disp) [m^2]
b3	1.17E-08	6.39E-13
b4	1.54E-08	5.14E-13
b5	1.48E-08	8.91E-12
b6	2.10E-08	7.17E-13

Table 4.9: Phantom experimental data mean noise for pressure and displacement measurements.

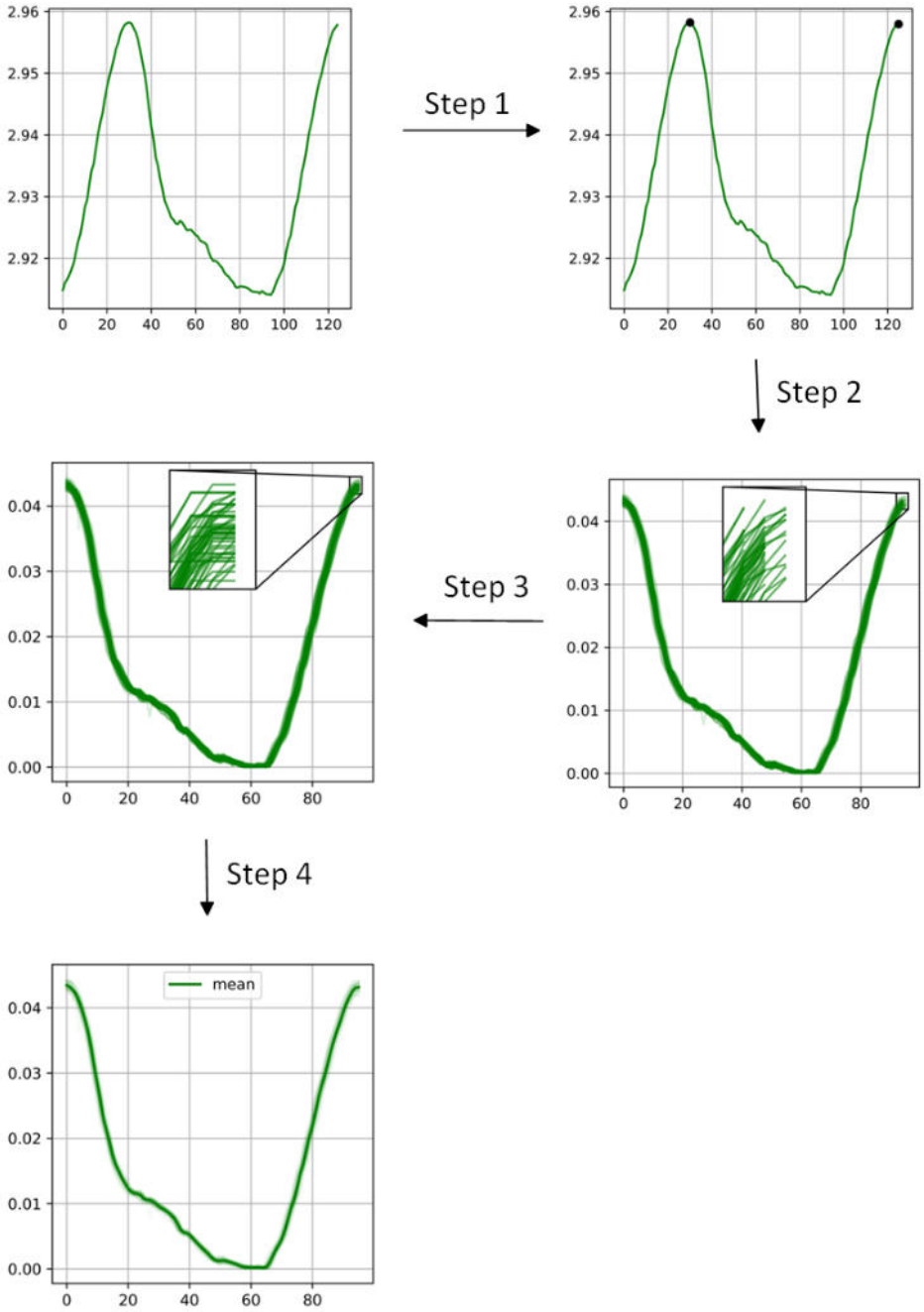


Figure 4.7: Data processing. Step 1 - identifying local maxima in the raw data. Step 2 - identifying each cycle. Step 3 - regularizing cycles lengths. Step 4 - Calculating the mean

Chapter 5

Model validation

In this chapter, two constitutive laws are tested on the phantom artery data. Two constitutive laws were tested, isotropic linear elastic and Neo-Hookean. Based on the value of the root mean square error (RMSE) between the simulated displacement and the measured displacement, the best constitutive law was chosen for the inverse problem.

Each of the four phantom samples: b3, b4, b5 and b6 has its own dedicated section where raw data are visually presented along with the averaged pressure and displacement cycle.

The averaged pressure cycle was used as a load on the inner phantom's wall in FEBio simulations. The averaged displacement curve was compared with the output of the numerical simulation.

Sample	Young's modulus	Model	RMSE [mm]
b3	engineering	Neo-Hookean	0.0007
b4	true	Linear	0.0019
b5	engineering	Neo-Hookean	0.0034
b6	true	Neo-Hookean	0.004

Table 5.1: Summarized results of the model validation.

The simulations were performed using an experimental Young's modulus (Tab. 4.6) taken from the uniaxial tensile test described in section 4.2. Two values were reported for each sample; one was engineering Young's modulus which is

related to the reference configuration, the other is true Young's modulus, which is related to the current (displaced) configuration. For better readability of this chapter, the results with the lowest RMSE are reported in table 5.1.

Based on the values in the Table 5.1 the best constitutive law based on the simulations is a Neo-Hookean model which gives the lowest root mean square error for three out of four cases. This model will be used as a governing model for the inverse problem of laboratory data and medical data.

The remainder of the chapter may be treated as a report on the simulations for each sample. This conclusion is proven in the following subsections of this chapter.

5.1 Sample b3

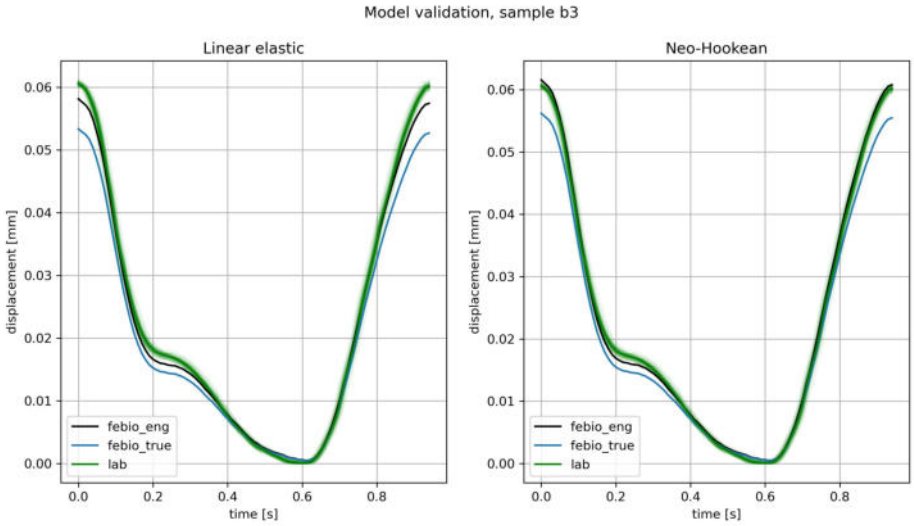


Figure 5.1: Model validation for sample b3 - linear elastic and Neo-Hookean for different values of Young's modulus. Neo-Hookean model with engineering Young's modulus fit the laboratory data the best. The shaded area behind the green curve (laboratory data) represents all measured displacement cycles measured by the camera.

Model	RMSE [mm] E_{eng}	RMSE [mm] E_{true}
Linear	0.0012	0.0035
Neo-Hookean	0.0007	0.0024

Table 5.2: Sample b3 root mean squared error. Engineering stress value and Neo-Hookean model are giving the best agreement with experimental data.

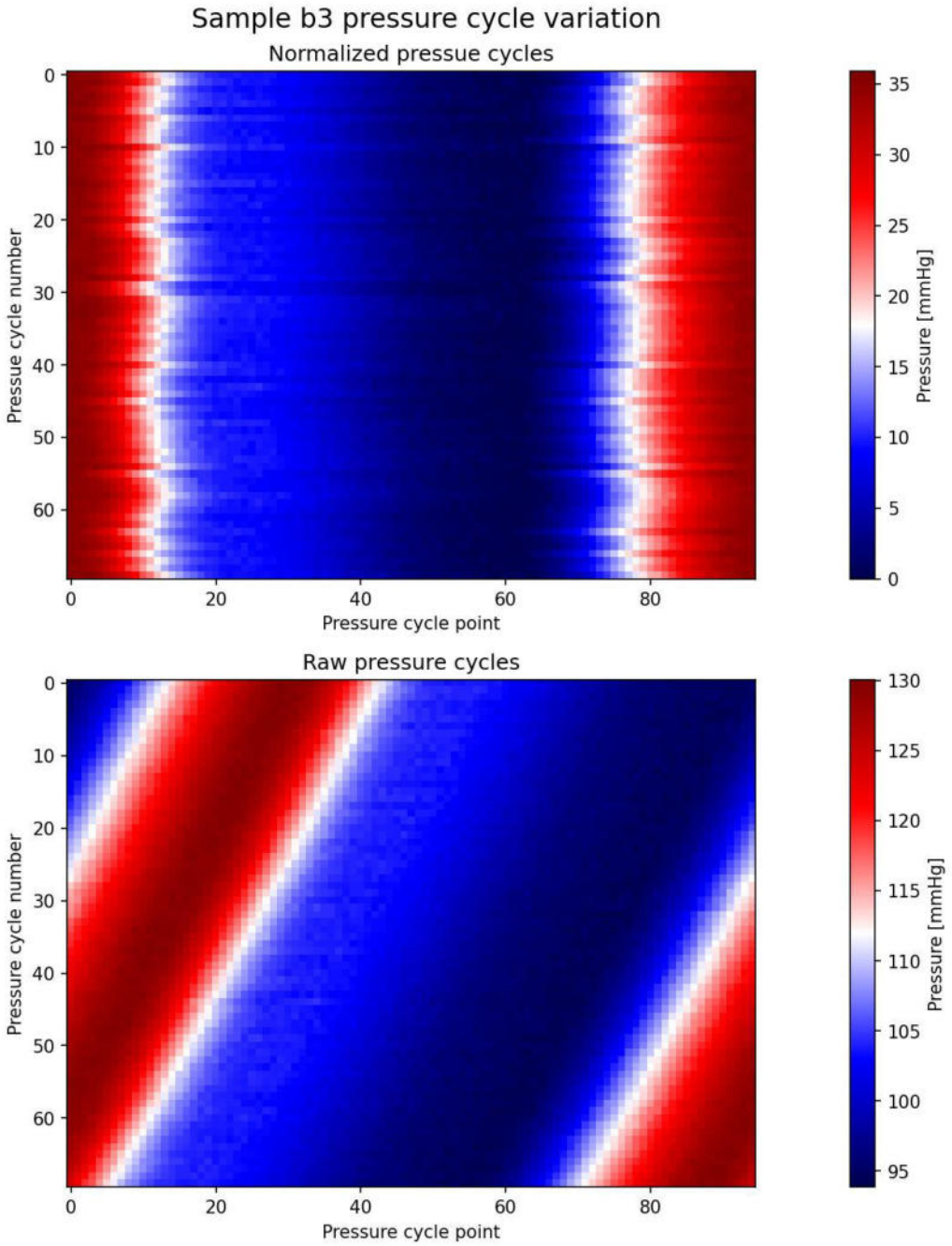


Figure 5.2: Pressure cycle variation for sample b3. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

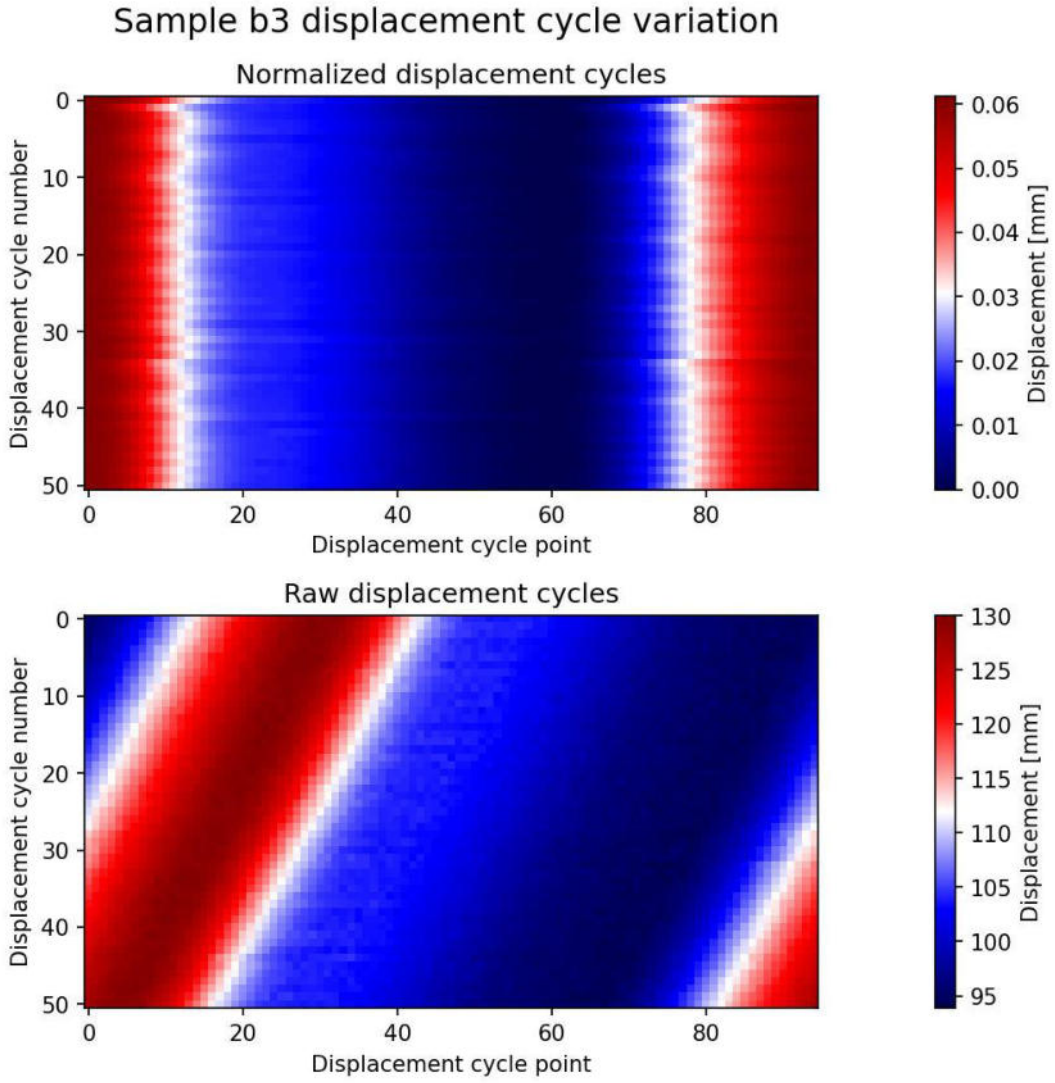


Figure 5.3: Displacement cycle variation for sample b3. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

5.2 Sample b4

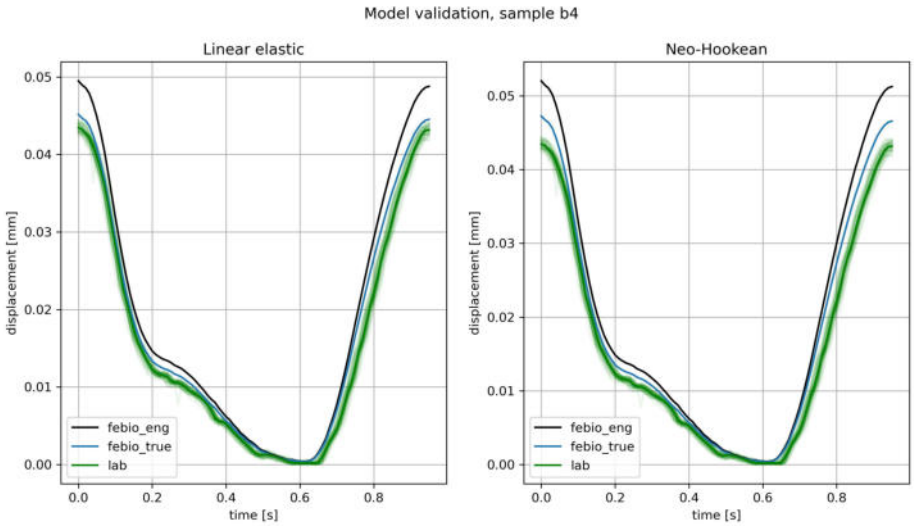


Figure 5.4: Model validation for sample b4 - linear elastic and Neo-Hookean for different values of Young's modulus. Linear elastic model with true stress value taken as a Young's Modulus, fits the laboratory data the best. The shaded area behind the green curve (laboratory data) represents all the measured displacement cycles measured by the camera.

Model	RMSE [mm] E_{eng}	RMSE [mm] E_{true}
Linear	0.0038	0.0019
Neo-Hookean	0.0047	0.0025

Table 5.3: Sample b4 root mean squared error. Engineering stress value and Neo-Hookean model are giving the best agreement with experimental data.

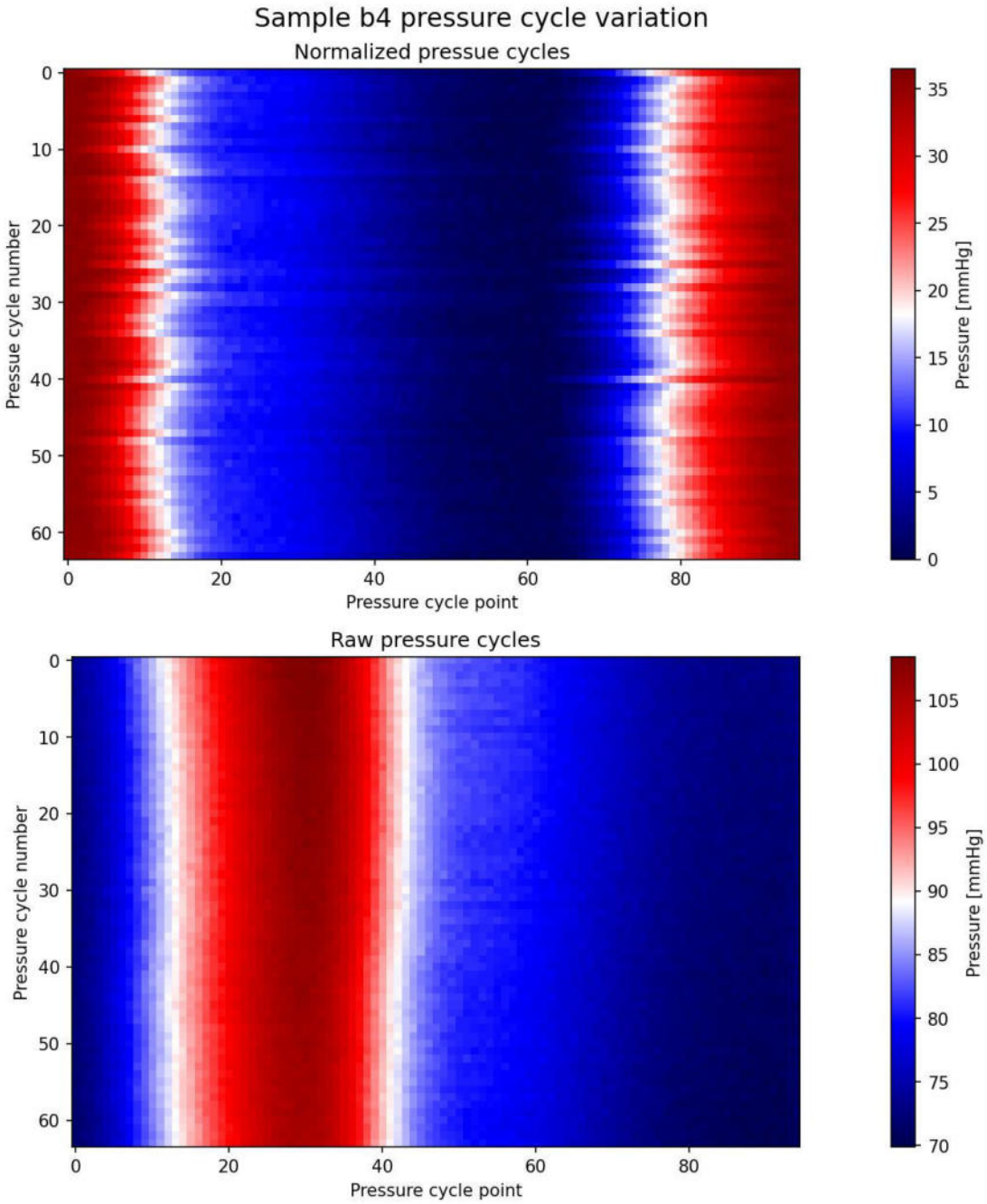


Figure 5.5: Pressure cycle variation for sample b4. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

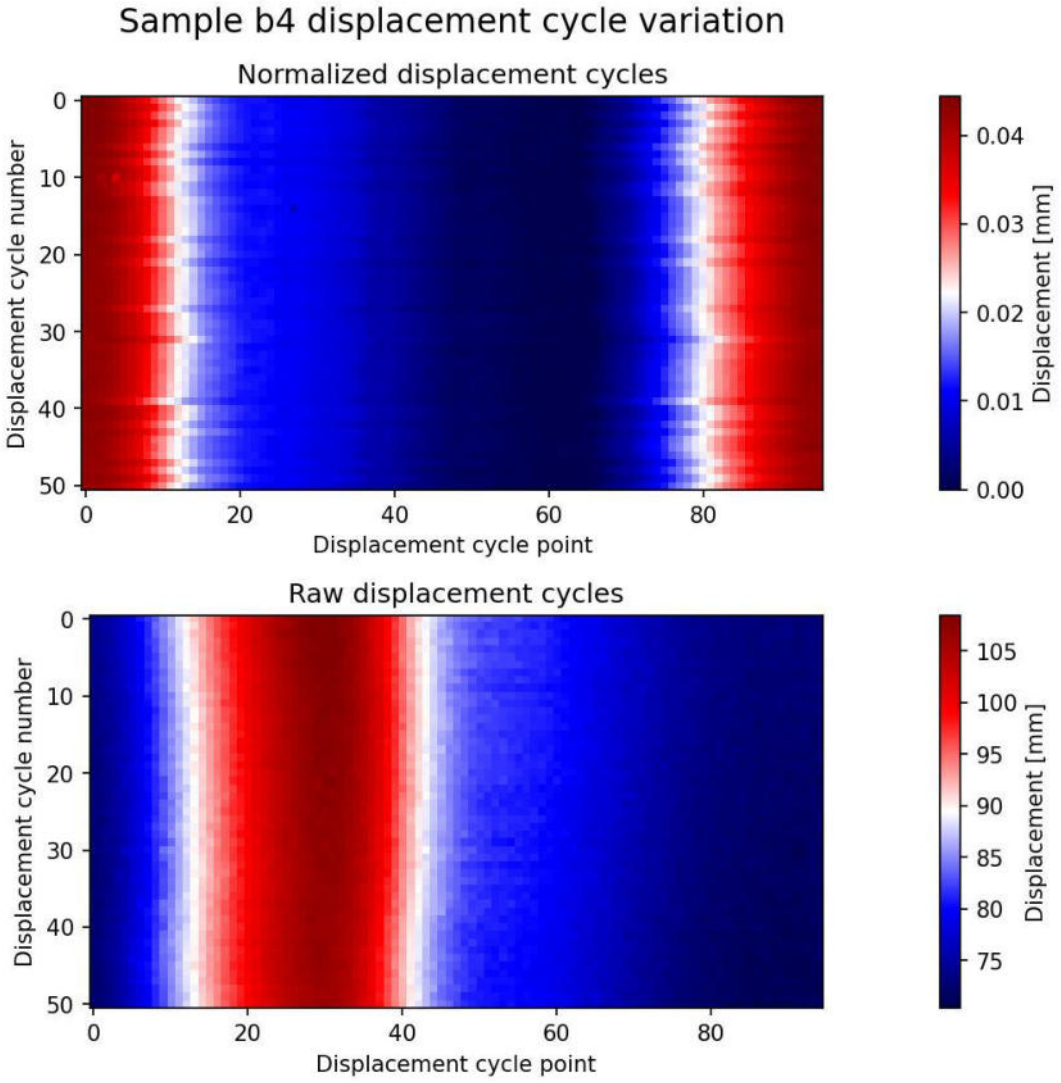


Figure 5.6: Displacement cycle variation for sample b4. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

5.3 Sample b5

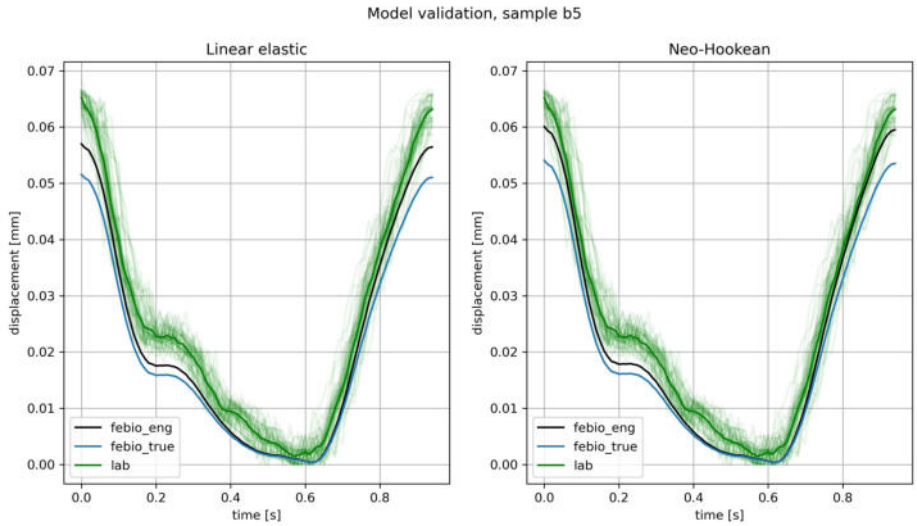


Figure 5.7: Model validation for sample b5 - linear elastic and Neo-Hookean for different values of Young's modulus. Neo-Hookean model with engineering Young's modulus fit the laboratory data the best. The shaded area behind the green curve (laboratory data) represents all measured displacement cycles measured by the camera.

Model	RMSE [mm] E_{eng}	RMSE [mm] E_{true}
Linear	0.0042	0.0067
Neo-Hookean	0.0034	0.0058

Table 5.4: Sample b5 root mean squared error. Engineering stress value and Neo-Hookean model are giving the best agreement with experimental data.

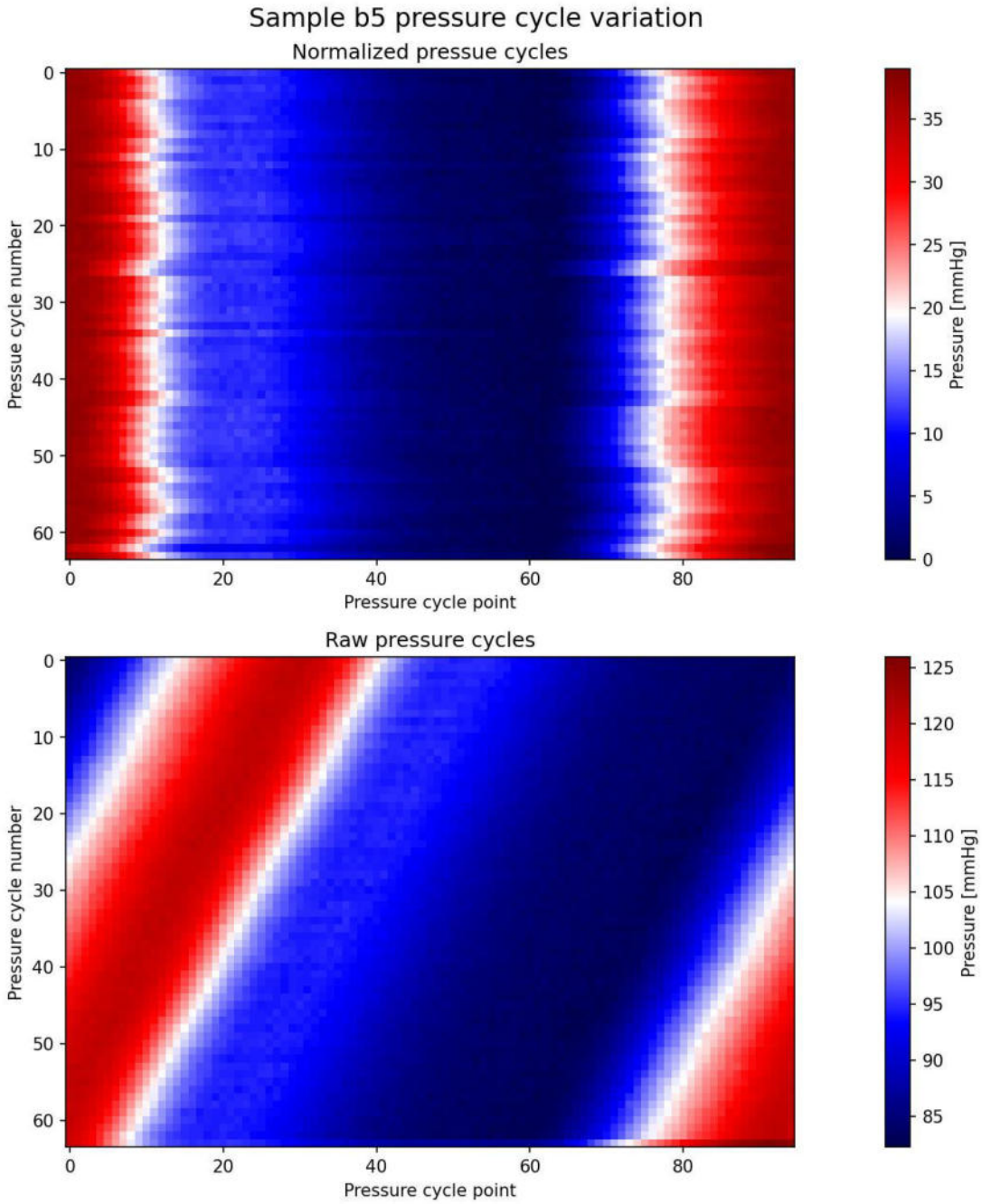


Figure 5.8: Pressure cycle variation for sample b5. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

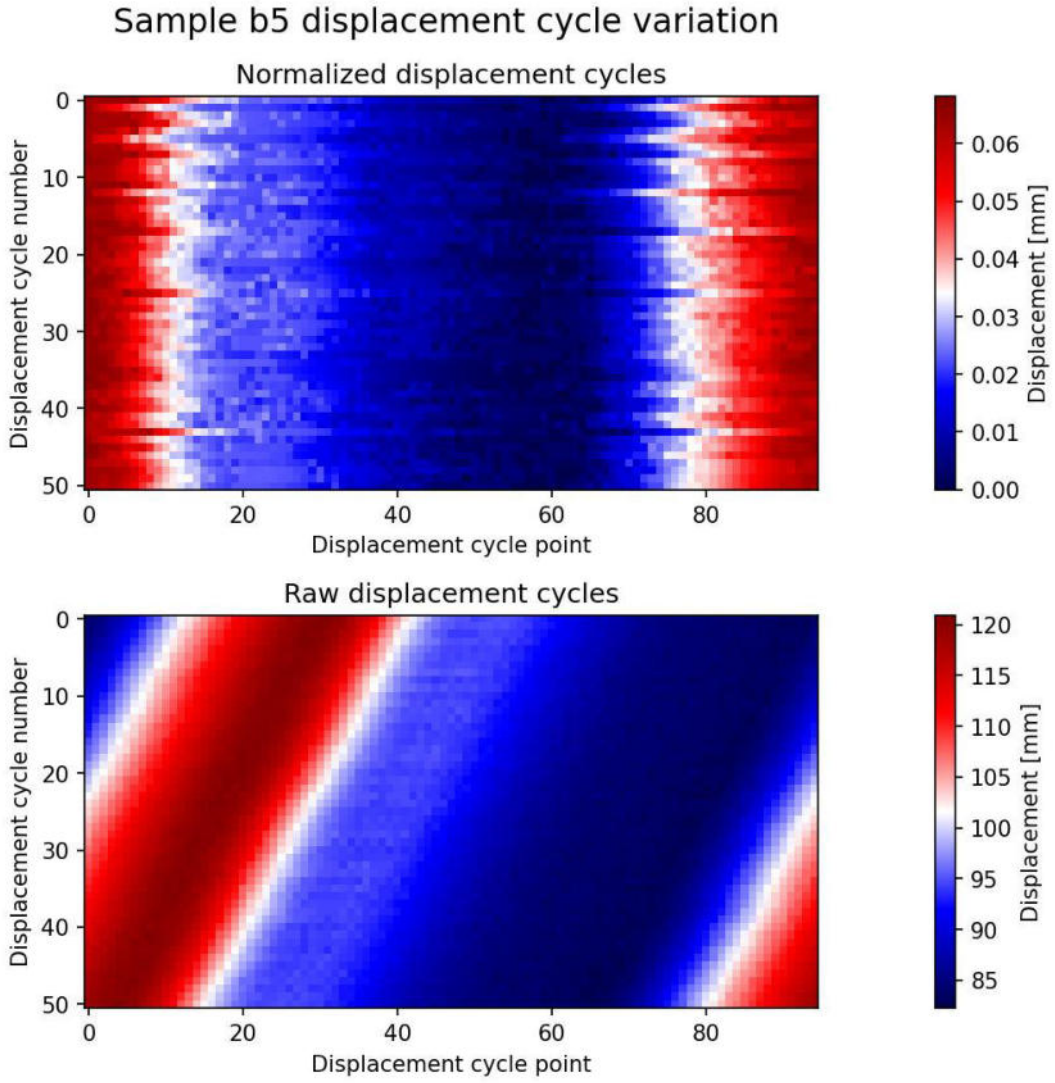


Figure 5.9: Displacement cycle variation for sample b5. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

5.4 Sample b6

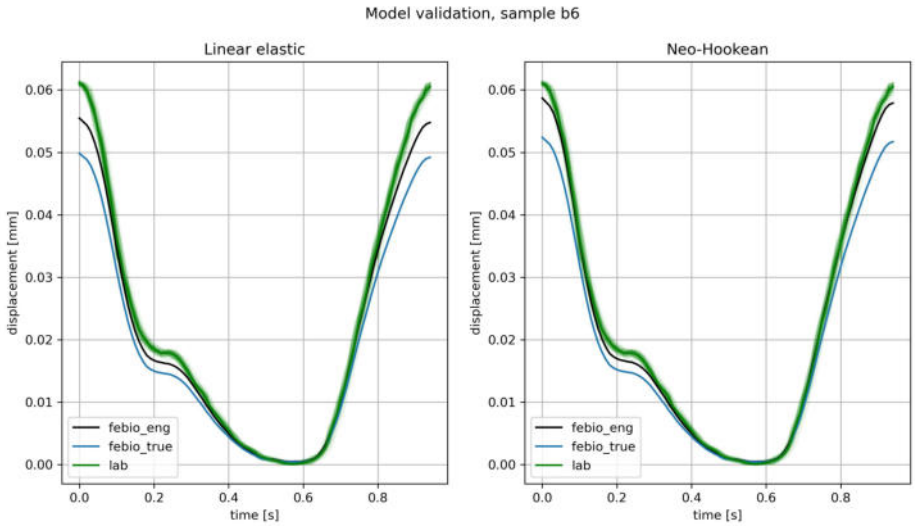


Figure 5.10: Model validation for sample b6 - linear elastic and Neo-Hookean for different values of Young's modulus. Neo-Hookean model with engineering Young's modulus fit the laboratory data the best. The shaded area behind the green curve (laboratory data) represents all measured displacement cycles measured by the camera.

	RMSE [mm]	RMSE [mm]
Model	E_{eng}	E_{true}
Linear	0.0023	0.005
Neo-Hookean	0.0011	0.004

Table 5.5: Sample b6 root mean squared error. Engineering stress value and Neo-Hookean model are giving the best agreement with experimental data.

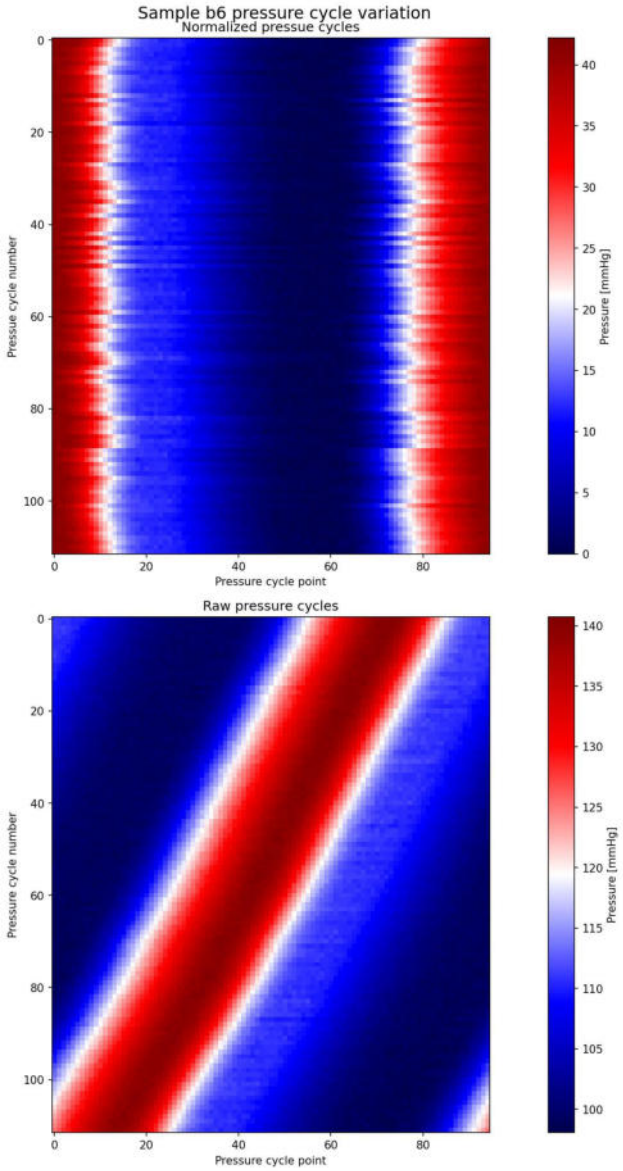


Figure 5.11: Pressure cycle variation for sample b6. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

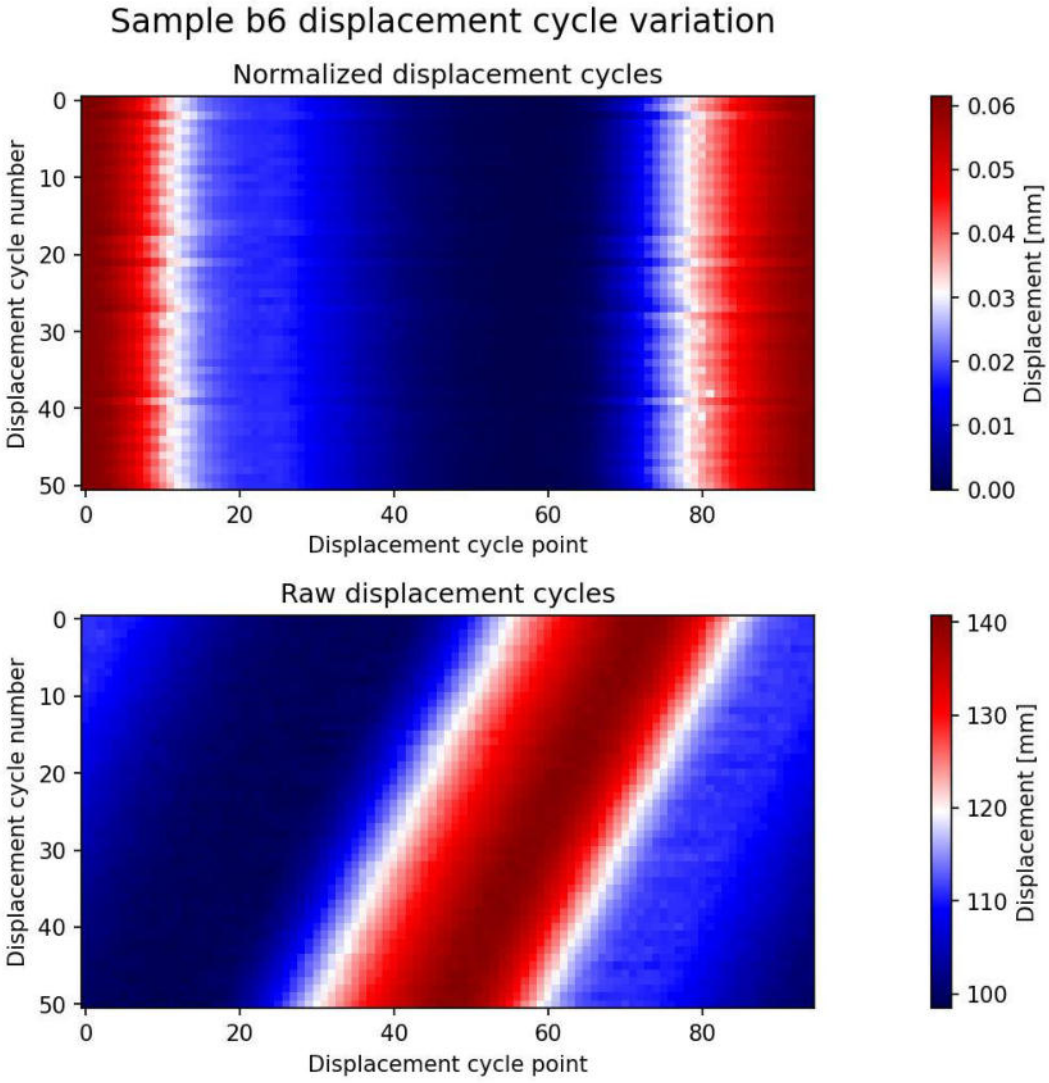


Figure 5.12: Displacement cycle variation for sample b6. The top figure shows the processed data. The bottom figure shows the raw data assuming constant length of the cycle.

Chapter 6

Numerical examples and results

6.1 Testing the algorithms on semi-artificial data

To test the extended Kalman filter and the dual extended Kalman filter, displacement data were generated using a thick-walled cylinder with fixed ends solution (Eq. 6.1).

$$u(R,t) = \frac{1+\nu}{E} \frac{r_{in}}{1-\left(\frac{r_{in}}{r_{out}}\right)^2} \left[\frac{r_{in}}{R} + (1-2\nu) \left(\frac{r_{in}}{r_{out}}\right)^2 \frac{R}{r_{in}} \right] p(t) \tag{6.1}$$

Noiseless pressure (averaged laboratory data) curve was used from previous research [77] and fed into the analytical equation above. Then Gaussian noise with known parameters was added independently to the pressure and displacement values. FEBio software was then used for the inverse problem procedure.

The constitutive law of the solid domain was set Isotropic Linear Elastic material. Geometry and Poisson's ratio was assumed to be known: inner radius 2.639 mm, outer radius 2.986 mm and Poisson' ratio equal to 0.4. The data was generated for $E = 1.4$ MPa. The initial guess for Young's modulus was set to 0.8 MPa. The rate of change of pressure was calculated using Savitzky-Golay filter with window length of 9 and polynomial order of 5. The time increment was equal to the laboratory time interval between each measurement - 0.01s.

Gaussian noise was then added to the data and Savitzky-Golay filter was used once again to find standard deviation of the noise. The recovered noise was

then used in Kalman filtering approach as input data. The true (controlled) noise and estimated noise are summarized in Tab. 6.1

Variable [Unit]	True std	Estimated std
p [MPa]	1.00E-04	9.19E-05
$\frac{dp}{dt}$ [Mpa/s]	1.00E-03	1.92E-03
u [m]	2.00E-06	1.80E-06

Table 6.1: Gaussian noise with known standard deviation was added to the noiseless data. Savitzky-Golay filter was used to estimate the noise in generated noisy data.

The transition matrix A in the extended Kalman filter for state variables is:

$$A = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & 0 \\ \left.\frac{df(E_{k-1}^{+})}{dp}\right|_{p=p_{k-1}^{+}} & 0 & 1 \end{bmatrix} \tag{6.2}$$

Where $f(E)$ is the Python function that runs the FEBio software. The state vector x is given as:

$$x_k^- = \begin{bmatrix} p_{k+1} \\ \dot{p}_{k+1} \\ u_{k+1} \end{bmatrix} \tag{6.3}$$

Where p is the pressure, $\dot{p} = \frac{dp}{dt}$ and u is displacement of the outer wall. The covariance matrix P for state variables is initialized as a unit matrix for this example and for every simulation from now on. The uncertainty in Young's modulus is set to 0.1. Matrix Q for state variables is set constant:

$$Q = \begin{bmatrix} \frac{dt^4}{4} & \frac{dt^3}{2} & 0 \\ \frac{dt^3}{2} & dt^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.4}$$

The uncertainty Q associated with Young's modulus is 1e-12 MPa. The following subsections summarize the data from the filtering approach and the plots are shown for the updated value of quantity of interest against the measurement data.

6.1.1 Testing Kalman filtering for parameter estimation

EKF is used to show the performance of the algorithm for Young's Modulus estimation only, state variables are neglected in prediction and update steps. The graphical solution is shown in Fig. 6.1. The tabularized data will be summarized later.

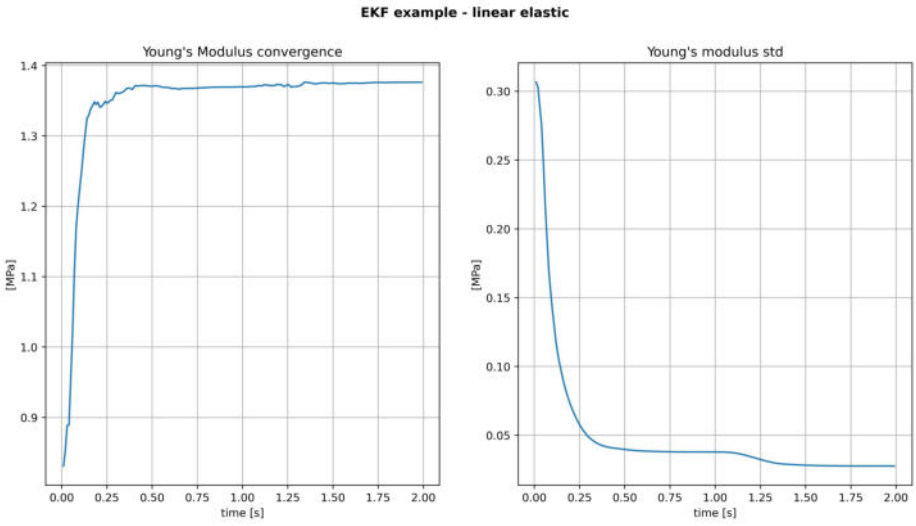


Figure 6.1: Example of the EKF approach for the estimation of Young's Modulus. Linear elasticity.

The Dual EKF predicts and updates both the state variable and the Young's modulus. The solution is plotted in Figures 6.2 and 6.3. Summarized solution for each method is found in the Tab. 6.2.

The Kalman filters both EKF and DEKF were initially run for 200 time steps, and the average standard deviation and young's modulus of the last 50 time steps were calculated. It reveals that the evaluation of only 150 time steps was necessary for the convergence. The computational time of these simulations was measured and is given in the results table 6.2.

The results and time efficiency were compared with more classical approach to the inverse problem. One cycle was evaluated with constant Young's modulus throughout the whole cycle. Once the cycle is finished, the calculated displacement curve is compared with the measurement. To minimize the cost function, another guess of Young's modulus is made, and the procedure continues until some convergence criteria are fulfilled.

For this technique, Scipy's implementation of Brent's method was used. One of its optional outputs is the number of function calls. In the case of Brent's method, the total number of function calls was equal to 1300 to fulfill the convergence criterion set to 0.001.

The cost function was:

$$\frac{\Delta u_s - \Delta u_m}{\Delta u_m} \tag{6.5}$$

Where Δu is calculated as:

$$\Delta u = max(u) - min(u) \tag{6.6}$$

which is the difference between maximum and minimum displacement for simulation results Δu_s and for measurement data Δu_m.

	EKF	DEKF	Brent
E0 [Mpa]	0.8	0.8	(0.5, 1.6)
E [Mpa]	1.378	1.383	1.383
std(E [Mpa])	0.028	0.02	-
No. of function calls	300	600	1300
Time [min]	3.5	7.5	10.5

Table 6.2: Comparison of computational cost and results between EKF, DEKF and Brent optimization method.

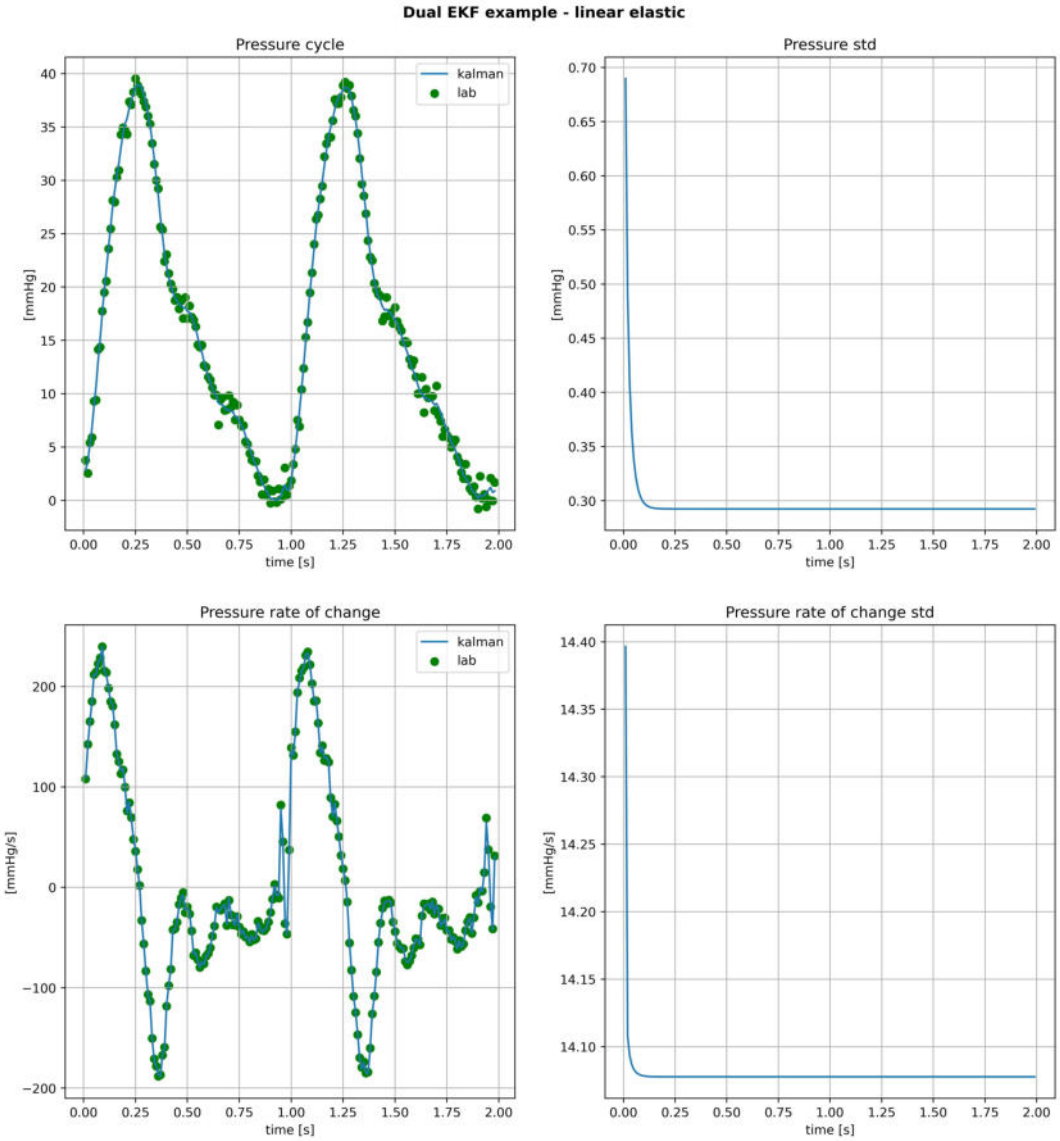


Figure 6.2: Example solution of the Dual EKF approach - pressure and pressure rate estimation.

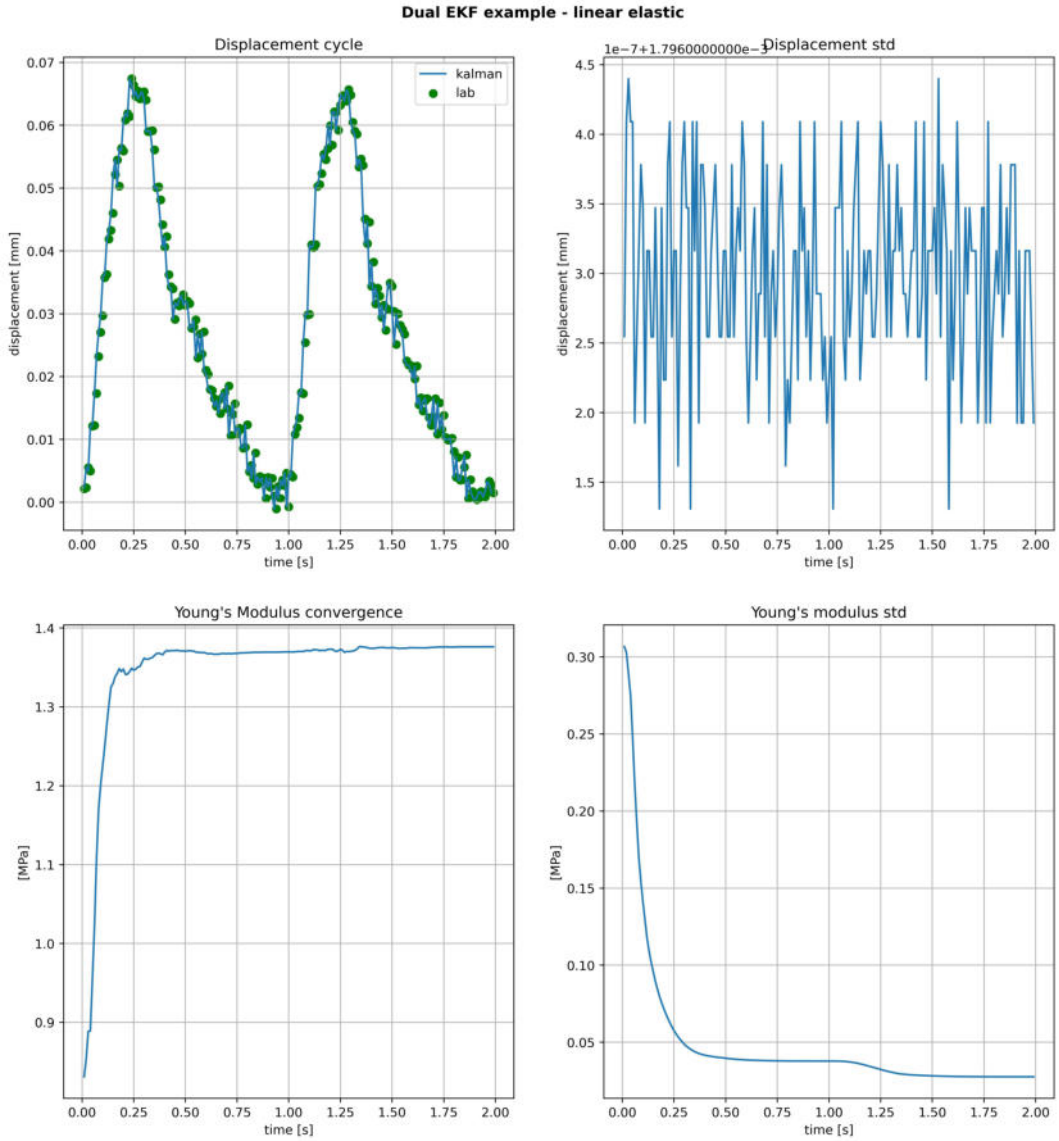


Figure 6.3: Example solution of the Dual EKF approach - displacement and Young's modulus estimation.

6.2 DEKF for the laboratory and medical data

6.2.1 Laboratory data

Dual Extended Kalman filter was chosen for the estimation of the Young's modulus of the four experimental materials.

The simple EKF approach where only Young's modulus is being estimated was not satisfactory due to convergence issues. The same issues arose when the raw data was used for both cases (DEKF, EKF), hence the averaged data was used as the measurement data.

Small oscillations occur in the estimated "stiffness", which can be explained by the shape of pressure and displacement waveforms. The linear elastic and Neo-Hookean models generate a displacement waveform which has the exact shape of the pressure waveform, which can be visually demonstrated by scaling both curves to the same range of values, for example between 0 and 1. The points corresponding to systolic pressure and the points nearby this peak in both displacement and pressure waveforms do not coincide, i.e. have different shapes.

For this reason the Young's Modulus value exhibits bumps, each associated with the data points in the vicinity of systolic pressure. For the rest of the cycle the plot of Young's modulus vs. time is flat. Figures Fig. 6.4 and Fig. 6.5 present the results for sample b3. Estimated pressure, rate of change of pressure and displacement are plotted against the experimental data, and the Young's modulus convergence is shown as the last. Each of these quantities has a corresponding standard deviation that varies with each time step and is plotted to the right of the quantity of interest.

The retrieved Young's modulus values for each sample are listed in the Tab. 6.3, where E_{kalman} is the result of the inverse problem averaged from time equal to 6s up to the end (10s), std(E_{kalman}) is the standard deviation associated with Young's modulus and finally E_{ref} is the reference value from mechanical testing yielding the smallest RMSE for the displacement.

The discrepancy between experimental stiffness and retrieved from the filtering approach has to follow from the fact that the samples are inhomogeneous. The thickness varies heavily, as was shown in Fig. 4.6 representing the thickness distribution of each sample.

In the literature, it has been reported that the Young's modulus is a function of

thickness, as the thickness decreases, the Young's modulus increases [90]. The simulations were run again with the new Young's moduli, and the displacement RMSE was calculated again and compared with the lowest RMSE corresponding to the experimental Young's modulus from uniaxial tensile testing.

The RMSE is summarized in the Tab. 6.4. The RMSE decreased for each sample, meaning that the values retrieved by the Dual Extended Kalman Filter are a better fit. The change for sample b3 is the smallest and the changes are of the order of 1/100000 mm.

Sample	E_{kalman} [Mpa]	std(E_{kalman}) [Mpa]	E_{ref} [Mpa]
b3	1.45	0.0015	1.43
b4	1.99	0.0022	1.82
b5	1.38	0.0017	1.50
b6	1.78	0.0018	1.84

Table 6.3: DEKF results of Young's Modulus E for the phantom samples. E_{kalman} - kalman result, std(E_{kalman}) - standard deviation of E, E_{ref} - best experimental fit.

Sample	$RMSE_{ref}$ [mm]	$RMSE_{kalman}$ [mm]
b3	0.0007	0.0007
b4	0.0019	0.0011
b5	0.0034	0.0024
b6	0.0011	0.0006

Table 6.4: RMSE for displacement cycle for the new Young's modulus and experimental for reference.

6.2.2 Medical data

Median value of Young's modulus of carotid artery is reported to be 613.97 kPa with standard deviation of 155.24 kPa in the case of subjects with no plaques [91]. For subjects with presence of plaques the reported values are higher, equal to 749.01 kPa (median) and 239.39 kPa (standard deviation) [91]. The resulted data of applying Dual Extended Kalman Filter for parameter estimation reported in the Tab. 6.5 fall into the reported interval and are physiologically possible values of the Young's modulus.

Patient	E [kPa]	std(E) [kPa]
patient1	582	0.1
patient2	461	0.2
patient3	837	0.07

Table 6.5: Young's modulus and its standard deviation, RMSE of the displacement for the medical data

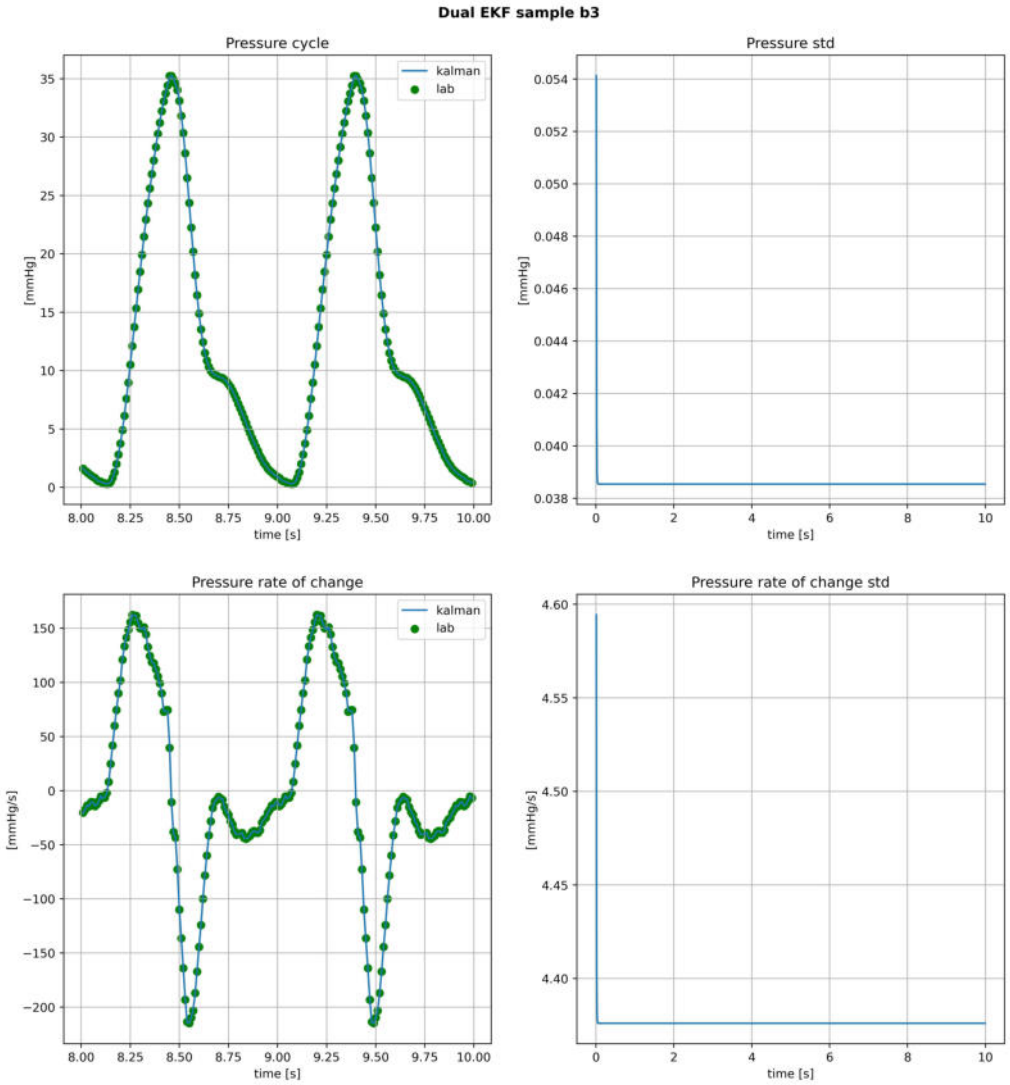


Figure 6.4: Pressure and pressure rate of change with associated standard deviation for sample b3.

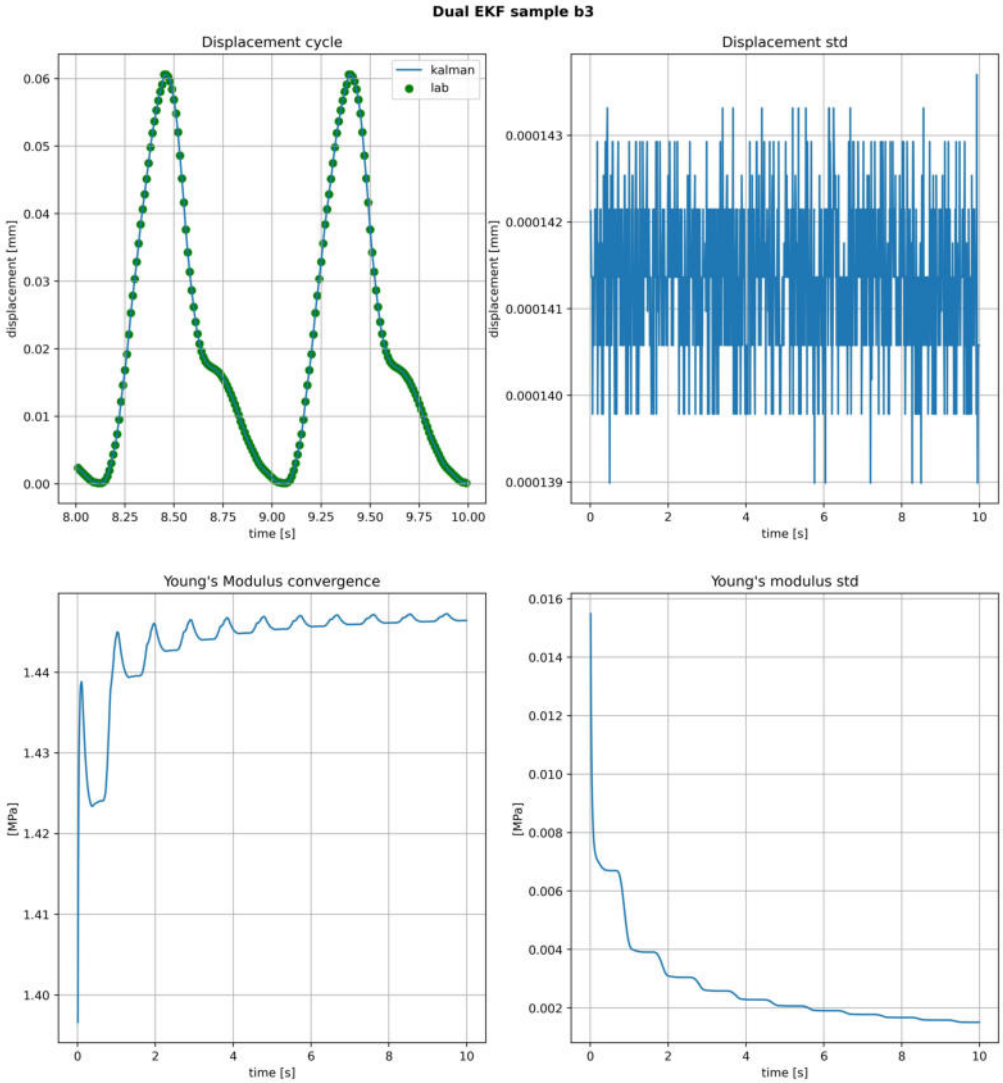


Figure 6.5: Displacement and Young's modulus with associated standard deviation for sample b3.

Chapter 7

Discussion and Conclusions

The left common carotid artery phantom thickness was varying along its length and perimeter. The average thickness was measured with an X-ray (Tab. 4.4, Fig. 4.6). The thickness distribution turned out to be in some cases a bimodal distribution.

Regardless of the difficulties, the model validation study performed in Chapter 5 revealed good agreement with the mechanical properties data from uniaxial testing. The ultrasound measurement of the phantom displacement could not reach its full potential due to wave reflections in the glass container and water in which the phantom was immersed. The decision for such medium and container material was dictated by the measurement procedure for which cameras were used to register the displacements. The paper by Sinek, Mesek et al. [77] offers a deeper explanation of the experimental rig and the results. In this paper, mixed effects were used to quantify the uncertainty in measurements.

The mean difference between the measured displacement by camera and ultrasound was 0.0113 mm. As expected, the uncertainty of the ultrasound measurement (0.03 mm) was higher than the uncertainty of the cameras (order of 1e-4 mm). The ultrasound and camera-derived displacements were in agreement thus confirming the experimental phantom and ultrasound to provide a reproducible experimental model for generating data for developing and testing new methods for non-invasive assessment of the common carotid artery.
The successful development of the experimental rig was necessary for the development of inverse problem procedure. The chosen method was based on extended Kalman filter, namely a dual extended Kalman filter was chosen.

This approach consists of two filters, one is filtering the state variables such as pressure and displacement. The second one is relying on the filtered data and estimates Young's modulus. The proposed method yielded results close to the experimentally determined stiffness, and the RMSE (for displacement) decreased in all cases. Mechanical experiments indicated Young's modulus of 1.43, 1.82, 1.50 and 1.84 Mpa. The value retrieved with Dual EKF gave results 1.45, 1.99, 1.38 and 1.75 MPa respectively.

For the medical data, the retrieved Young's modulus is found to be within range of 582 - 837 kPa which falls into physiological range (613 $\pm$ 155.24 - 988 $\pm$ 239.39 kPa) found in literature, which clearly shows the potential of applying proposed methodology for real-time estimation.
The basic filtering approach may be used for the estimation of unknown model parameters. In real applications, the measurements should be carefully investigated for the noise distribution.

In the case of the experimental rig the pressure transducer noise wasn't Gaussian, the distribution was narrower with a higher peak compared to the Gaussian distribution [77]. This suggests that using an unscented Kalman filter does not require derivatives of the model, but the matrices size may grow fast due to their augmentation to incorporate the noise of each variable inside the matrix.

For a successful implementation of the proposed technique, it is required to build a reduced order model of the artery instead of relying on external software at each time step. This would make the simulations faster and the proposed approach could be used in real-time.

A potential bottleneck for real-time assessment of the compliance or Young's modulus could be image processing of the ultrasound measurement. The determination of wall thickness requires special care due to blurred image and the difficulty in detecting the edges of the vessel. The Kalman filter could be augmented to estimate the thickness in addition to the stiffness. In that case, more dynamic equations should be modeled by the transition matrix.

Volumetric flowrate and velocity could be measured by Doppler ultrasound, then the inner diameter could vary until the volumetric flowrate matches the measured value

$$\dot{V} = vA \tag{7.1}$$

Where $\dot{V}$ is the volumetric flowrate, v is the blood velocity and A is the lumen cross-section.

Two way fluid-structure interaction simulation could be performed using FEBio software since it is easy to automate and the software offers a coupled solver. Executing external software each time step inside the Kalman filter algorithm would be too time-consuming, therefore a reduced order model of the dynamics would be highly recommended for such an approach.

Parallelization of the Kalman filter could be considered if one thinks of real-time procedure [92].

Chapter 8

Appendix

8.1 Working principle of Kalman filter

The behind the scenes of Kalman filter are best described and visualized if one thinks of it in terms of Bayesian filtering. The Bayesian filter is a recursive filter and is summarized with two equations:

$$\overline{bel}(x_k) = \int p(x_k|x_{k-1})bel(x_{k-1})dx \tag{8.1}$$

$$bel(x_k) = \eta p(z_k|x_k)\overline{bel}(x_k) \tag{8.2}$$

In eq.8.1 the term $\overline{bel}(x_k)$ represents the predicted probability density function (prior) which is then corrected using measurement with eq.8.2. The updated belief (posterior) is not a proper probability density function since it does not sum to 1. To make it sum to 1, a constant η needs to be found. The integrals in above equations are usually intractable, however if it is assumed that the PDFs are normal and the dynamic system is linear, a closed form solution can be derived yielding a Kalman filter [93, 65].
The following example simulates an object moving with nearly constant velocity. The velocity variation is caused by some external factors like drag force or sensor inaccuracy. The velocity is believed to be around 1 m/s. The belief can be described by describing the velocity as a normal distribution centered at 1 and some variance associated with this value. Let's assume the velocity $v \sim N(1, 1)$. Where N is a normal distribution with parameters mean μ and variance σ^2, $N(\mu, \sigma^2)$. At the beginning it is believed that the object is situated

somewhere between 0 and 10 m, $N \sim (5, 2)$. These parameters constitute an initial guess about the system. The position of the object will be updated with each time step equal to 1 s. The prediction of position x will be performed using simple dynamic model:

$$x_k^- = x_{k-1}^+ + v\Delta t \tag{8.3}$$

Since the time step equals 1, the above equation simplifies to:

$$x_k^- = x_{k-1}^+ + v \tag{8.4}$$

Substituting the Normal distributions for each variable, we arrive with summation of normal distributions. It has to recalled that a mixture of normal distributions is also a normal distribution [94]:

$$N(\mu_k^-, \sigma_k^{2-}) = N(5, 2) + N(1, 1) \tag{8.5}$$

For the sum of normal distributions there fortunately exists an analytical formula to evaluate new mean and variance of the new distribution [94]:

$$\mu = \mu_1 + \mu_2 \tag{8.6}$$

$$\sigma^2 = \sigma_1^2 + \sigma_2^2 \tag{8.7}$$

So the new mean and new variance of a normal distribution is just the sum of the the means and variances of the the two old normal distributions. It follows from the two above equations that the mean is just shifted on the x axis and what is more important, the uncertainty in the prediction step increases. The visual representation of this phenomenon is shown in the Fig. 8.1. The initial position (blue curve) is updated with equation Eq.8.4. The result is shifted to the right by one unit, and the spread of the distribution is wider than the spread of the initial guess. The predicted position in the Bayesian approach is often referred to as a prior [95].

The update step is a data assimilation step. In this step the belief about a position of the object is updated yielding posterior probability distribution function. The Bayes theorem states [95]:

$$p(x|z) = \frac{p(z|x) \cdot p(x)}{p(z)} \tag{8.8}$$

where:

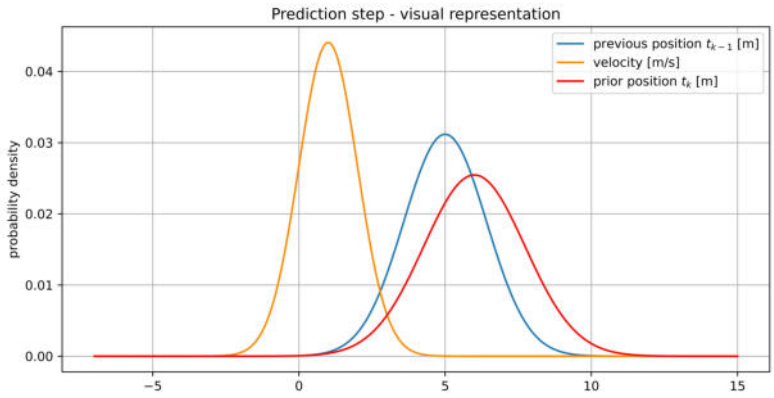


Figure 8.1: Graphical representation of the prediction step in Kalman filtering. The blue curve may represent here either the initial guess or the posterior (updated belief about position), the orange curve represents the belief in the velocity of the object and the red curve is a result of summation of the two normal probability distribution functions. The result is the prediction (prior) of the system at the next time step t_k.

- $p(x|z)$ - conditional probability of position of the object x given measurement (measured position) z
- $p(z|x)$ - likelihood of observing the data z under hypothesis that the object position is described by the probability density function p(x)
- $p(z)$ - marginal probability of getting the measurement z

The other form of this equation is also given by:

$$p(x_i|z) = \frac{p(z|x_i) \cdot p(x_i)}{\sum_{i=1}^{n} p(z|x_i) \cdot p(x_i)} \tag{8.9}$$

which is a discrete form of Bayes' Theorem. The continuous version:

$$p(x|z) = \frac{p(z|x) \cdot p(x)}{\int_{-\infty}^{\infty} p(z|x) \cdot p(x) dx} \tag{8.10}$$

The equation 8.10 is often rewritten as a proportionality since the denominator is a constant scaling the numerator so that the posterior sums to 1. This form is used if the integral is intractable and in that case Markov Chain Monte Carlo method is used to obtain the posterior distribution [95].

$$p(x|z) \propto p(z|x) \cdot p(x) \tag{8.11}$$

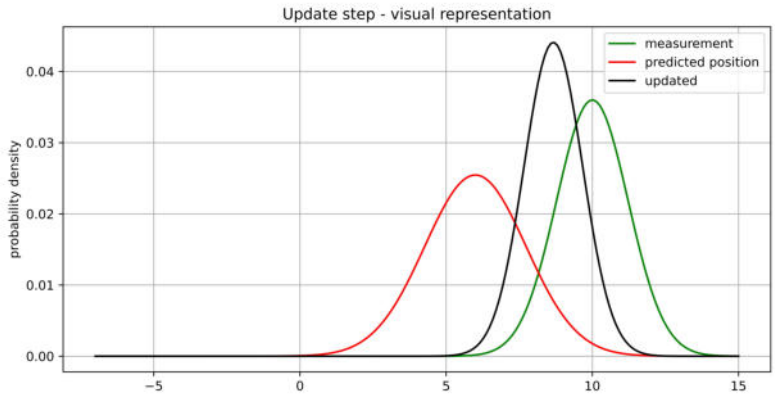


Figure 8.2: Graphical representation of the update step in Kalman filtering. The red curve represents the prior, green curve represent the measurement and the black curve is the posterior (updated belief about the position of the object). The posterior PDF has smaller standard deviation compared to the measurement and prior, meaning that confidence about the position was gained.

If the choice of prior and likelihood is chosen to be normal there exists a closed form solution for the posterior distribution, given by [94, 96]:

$$\mu = \frac{\sigma_1^2 \mu_2 + \sigma_1^2 \mu_1}{\sigma_1^2 + \sigma_2^2} \tag{8.12}$$

$$\sigma^2 = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2} \tag{8.13}$$

Given the measurement to be normally distributed random variable $N \sim (10, 1.5)$, the prior can be updated using the above equations. The resulting mean is between the prior and measurement and a level of certainty in the updated position is gained since the spread of the posterior is smaller compared to prior and measurement. It can be expected then that as the filter progresses with time, the uncertainty (standard variation/variance) will become lower with each iteration. The update step is illustrated in the Fig.8.2.

With each measurement, the new posterior probability becomes a prior to the next analysis. The described recursive process visualized was repeated for further artificial measurements to show the behavior of the Bayesian filter (Fig.8.3). The standard deviation indeed decreases with each measurement,

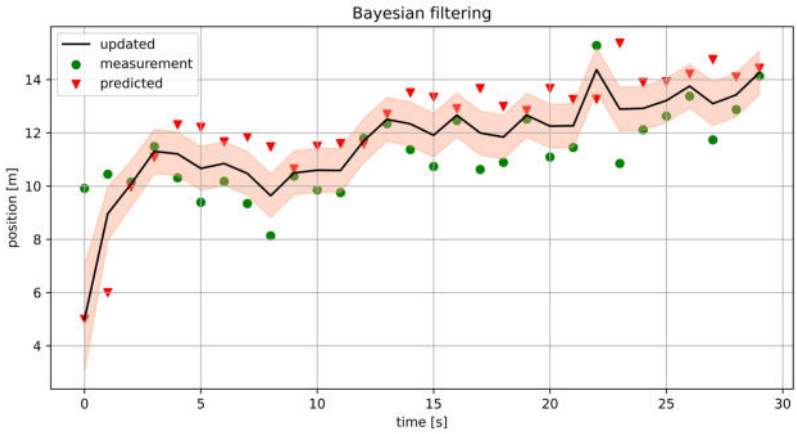


Figure 8.3: An output of the Bayesian filter (black line) against measured data points (green) and predicted predicted positions (red triangles). The shaded area represent the uncertainty in the filter output. It is calculated as the $output \pm \sigma^2$.

for better visualization, the standard deviation was plotted separately in the Fig.8.4. It was shown that Kalman filter can be derived from Bayesian filter and it is a good educational representation of the principles of Kalman filtering [97].

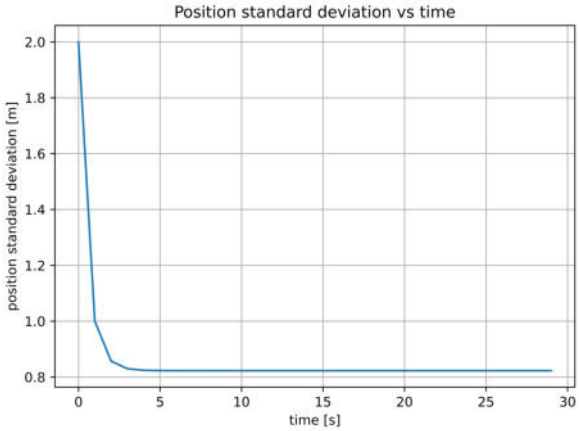


Figure 8.4: Varying standard deviation of the posterior probability of the object's position.

8.2 Data processing - Python code

Data processing function:

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from collections import defaultdict
from scipy.signal import find_peaks
from scipy import interpolate

def mean_data(data):
    # Find maxima
    peaks, _ = find_peaks(data, distance = 90)

    # Slice data into cycles
    sliced = defaultdict()
    count = 0
    for i in range(1,len(peaks)):
        start = peaks[i-1]
        stop = peaks[i]
        minimum = np.min(data[start:stop])
        sliced[count] = data[start:stop] - minimum
        sliced[count][-1] = sliced[count][0]
        count += 1

    # Mean period
    period = []
    for i in range(1, len(peaks)):
        period.append(peaks[i] - peaks[i-1])
    period = np.asarray(period)
    mean_period = np.mean(period)/100
    round_period = np.round(mean_period,2)

    # Regularize varying cycles into equal sized
        cycles
    time = np.linspace(0, round_period,\
    int(round_period*100))
    regularized = defaultdict()
```

```
35      for key in sliced.keys():
36          x = np.linspace(0, period[key]/100, period[key
                ])
37          y = sliced[key]
38          fill_value = y[0]
39          f = interpolate.interp1d(x,y,\
40          fill_value = 'extrapolate')
41          ynew = f(time)
42          ynew[-1] = fill_value
43          regularized[key] = ynew
44      n_cycles = len(regularized) # rows
45
46
47      # Returns cycles dictionary, no of rows,\
48      #no of columns
49      return regularized, n_cycles, int(round_period
            *100)
```

Plotting function:

```
1 def datamap(dictionary, rows, cols):
2      datamatrix = np.zeros((rows, cols))
3      for key in dictionary.keys():
4          datamatrix[key] = dictionary[key]
5      plt.figure(figsize = (10,3))
6      plt.imshow(datamatrix, cmap = 'seismic')
```

8.3 EKF and DEKF - Python code

```
1 import numpy as np
2 import os
3 import subprocess
4 import xml.etree.ElementTree as ET
5 import matplotlib.pyplot as plt
6 from IPython.display import display, clear_output
7
8 class Params:
9      # This class initializes outer, inner radius and
           Poisson's ratio for the analytical model for
```

```
        displacement of the wall of thick walled
10      # cylinder and it returns displacement of the
            outer wall
11      def __init__(self, b, a, ni):
12          self.b = b # outer radius in m
13          self.a = a # inner radius in m
14          self.R = b # radius to track in m
15          self.ni = ni # Poisson's ratio
16          self.const = (1+self.ni)*(self.a/(1-(self.a/
                self.b)**2))*(self.a/self.R +
17          (1-2*self.ni)*((self.a/self.b)**2) * self.R/
                self.a)
18      def disp(self,E,p):
19          return 1/E*self.const*p
20
21  class EKF(Params):
22      # This class is an implementation of Extended
            Kalman Filter and Dual Extended Kalman Filter
            for the analytical model of thick walled
            cylinder
23      # This class enherits from class Params,
            additionally it takes the Young's Modulus
            initial guess E_plus in MPa to fully describe
            the geometry
24      # and mechanical properties
25      # P_plus, Q_E, Q_L - parameters controlling
            behavior of the extened kalman filter
26      # P_plus - variance of the estimate Young's
            modulus
27      # Q_E - variance of the process noise
28      # Q_L - additional parameter controlling stability
              of calculating the Kalman Gain L
29      # xhat - initial guess of pressure, pressure rate
            of change dp/dt and displacement of the outer
            wall
30      # noise - estimate of the measurement noise for
            each quantity
31
```

```
32      def __init__(self, param, E_plus = 1.2, P_plus =
            0.1, Q_E = 1e-8, Q_L = 1e-10,
33                   xhat = [0.009, 7e-3, 9.7e-5], dt =
                         0.01, noise = [1e-5, 5e-4, 3e-6]):
34          super().__init__(param.b, param.a, param.ni)
35                           #param.P_state, param.P_E,
                                 param.Q_E_der, param.Rk,
                                 param.const)
36          self.E_plus = E_plus
37          self.P_plus = P_plus
38          self.Q_E = Q_E
39          self.Q_L = Q_L
40          self.xhat = np.array([xhat[0], xhat[1], xhat
                [2]])
41          self.dt = dt
42          self.P_state = np.eye(3)
43
44          self.Q_state = np.eye(3)
45          self.Q_state[0,0] = (dt**4)/4
46          self.Q_state[0,1] = self.Q_state[1,0] = (dt
                **3)/2
47          self.Q_state[1,1] = dt**2
48
49          self.Rs = np.eye(3)
50          self.Rs[0,0], self.Rs[1,1], self.Rs[2,2] =
                noise[0]**2, noise[1]**2, noise[2]**2
51
52      # C numerical
53      def central(self,E, p, h = 1e-5):
54          # returns numerical derivative with respect to
                 Young's modulus by central finite
                difference method
55          return (self.disp(E+h, p) - self.disp(E-h, p))
                /(2*h)
56
57      # C analytical
58      def analytical(self, E, p, h = 1e-5):
59          # returns analytical derivative of the
```

```
            analytical model
60          return -1/(E**2)*self.const*p
61
62
63      # def F_numerical
64      def F_central(self, E, p, h = 1e-5):
65          # returns a numerical value for the transition
                state matrix - governing dynamic recursive
                model of the system
66          return (self.disp(E, p+h) - self.disp(E, p))/h
67
68      # def F_analytical
69      def F_analytical(self, E, p, h = 1e-5):
70          # returns an analytical value for the
                transition state matrix - governing dynamic
                recursive model of the system
71          return 1/E*self.const
72
73      def evaluate(self, derivative_method, p, z):
74          # returns the updated Young's modulus,
                estimated displacement and variance of
                Young's modulus
75          # It takes the derivative_method input (string
                ) to choose method of calculation of the
                derivative
76          # p - pressure
77          # z - experimental measurement
78          # This function can be used to use Extended
                Kalman Filter for paramter estimation only
                (Young's modulus)
79          derivative_functions = {
80              'analytical': self.analytical,
81              'central': self.central
82          }
83
84          derivative_function = derivative_functions.get
                (derivative_method)
85
```

```
86          if derivative_function is None:
87              raise ValueError("Invalid derivative
                    method specified")
88
89          E_minus = self.E_plus
90          P_p = self.P_plus + self.Q_E
91          z_est = self.disp(E_minus, p)
92          #z_est = f(E_minus, p, param.const)
93          residual = z - z_est
94          C = derivative_function(E_minus, p, h=1e-10)
95          L = P_p*C*(1/(C*P_p*C + self.Q_L)) #z_est
                **2*(1e-7))) #
96          self.E_plus = E_minus + L*residual
97          self.P_plus = (1 - L*C)*P_p
98          return self.E_plus, z_est, self.P_plus
99
100     def DEKF(self, method, z):
101         # This function is used to update the belief
                in pressure, pressure rate dp/dt and
                displacement, it is already coupled with
                the function
102         # evaluate which gives the Dual Extended
                Kalman Filter
103         F_methods = {
104             'F_analytical': self.analytical,
105             'F_central': self.central
106         }
107
108         F_method = F_methods.get(method)
109
110         if F_method is None:
111             raise ValueError("Invalid method specified
                    ")
112
113         xpred = np.array([self.xhat[0] + self.xhat[1]*
                self.dt, self.xhat[1], 1/self.E_plus*self.
                xhat[0]])
114         #F = np.array([[1, self.dt,0],[0,1,0],[1/self.
```

```
            E_plus*self.const, 0, 0]])
115         F = np.array([[1,self.dt,0],[0,1,0],[F_method(
                self.E_plus, self.xhat[0]), 0, 0]])
116         Ppred = F@self.P_state@F.T + self.Q_state
117         H = np.eye(3)
118         S = H@Ppred@H.T + self.Rs
119         K = (Ppred@H.T)@np.linalg.pinv(S)
120         self.xhat = xpred + K@(z - H@xpred)
121         self.P_state = (np.eye(3) - K@H)@Ppred
122
123         e = self.evaluate(method[2:],xpred[0], z[2])
124         #est_e = self.evaluate(method[2:],xpred[0], z
                [2])[0]
125         est_e = e[0]
126         var_e = e[1]
127
128
129         return self.xhat, est_e, var_e, self.P_state
130
131
132 class EKF_febio():
133     # This class evaluates EKF and DEKF and the
            analytical model is replaced by Finite Element
            open-source software FEBio
134     # Suitable for isotropic elastic and Neo-Hookean
            model
135     def __init__(self, filepath, E_plus = 1.2, P_plus
            = 0.1, Q_E = 1e-8, Q_L = 1e-10,
136                  xhat = [0.009, 7e-3, 9.7e-5], dt =
                         0.01, noise = [1e-5, 5e-4, 3e-6]):
137
138         self.file = filepath
139         self.tree = ET.parse(self.file)
140         self.parent = os.path.dirname(self.file)
141         self.result = os.path.join(self.parent, 'jobs'
                , 'outer.csv')
142         #print(str(self.file))
143         self.E_plus = E_plus
```

```
144         self.P_plus = P_plus
145         self.Q_E = Q_E
146         self.Q_L = Q_L
147 
148         self.xhat = np.array([xhat[0], xhat[1], xhat
                [2]])
149         #self.xhat[0,0], self.xhat[1,1], self.xhat
                [2,2] = xhat[0], xhat[1], xhat[2]
150         self.dt = dt
151         self.P_state = np.eye(3)
152 
153         self.Q_state = np.eye(3)
154         self.Q_state[0,0] = (dt**4)/4
155         self.Q_state[0,1] = self.Q_state[1,0] = (dt
                **3)/2
156         self.Q_state[1,1] = dt**2
157 
158         self.Rs = np.eye(3)
159         self.Rs[0,0], self.Rs[1,1], self.Rs[2,2] =
                noise[0]**2, noise[1]**2, noise[2]**2
160 
161     def disp(self, E, p):
162 
163         d = self.tree.find('./Loads/surface_load/
                pressure')
164         p = p*1e6
165         d.text = str(p)
166         e = self.tree.find('.Material/material/E')
167         E = E*1e6
168         e.text = str(E)
169         self.tree.write('cyl.feb', encoding='iso
                -8859-1')
170         febio = subprocess.Popen('cyl.feb', shell=True
                )
171         febio.wait()
172         #outer = pd.read_csv(self.result,
                delim_whitespace=True)
173         outer = pd.read_csv('outer.csv',
```

```
                delim_whitespace=True)
174             outer_est = float(outer.iloc[-1]['='])
175             #outer_est = float(outer.iloc[6][1]) #- shift
176             return outer_est
177
178             # C numerical
179         def central(self,E, p, h = 1e-5):
180             return (self.disp(E+h, p) - self.disp(E-h, p))
                    /(2*h)
181
182             # def F_numerical
183         def dfdp(self, E, p, h = 1e-5):
184             return (self.disp(E, p+h) - self.disp(E, p))/h
185
186         def evaluate(self, p, z):
187             # EKF implementation for estimating Young's
                    modulus
188             E_minus = self.E_plus
189             P_p = self.P_plus + self.Q_E
190             z_est = self.disp(E_minus, p)
191             #z_est = f(E_minus, p, param.const)
192             residual = z - z_est
193             C = self.central(E_minus, p, h=1e-10)
194             L = P_p*C*(1/(C*P_p*C + self.Q_L))# z_est
                    **2*(1e-1)**2))
195             self.E_plus = E_minus + L*residual
196             self.P_plus = (1 - L*C)*P_p
197             return self.E_plus, self.P_plus
198
199         def DEKF(self, p, z):
200             # Together with method "evaluate" it is an
                    implementation of Dual Extended Kalman
                    Filter
201             xpred = np.array([self.xhat[0] + self.xhat[1]*
                    self.dt, self.xhat[1], self.disp(self.
                    E_plus, self.xhat[0])])
202             #F = np.array([[1, self.dt,0],[0,1,0],[1/self.
                    E_plus*self.const, 0, 0]])
```

```
203         F = np.array([[1,self.dt,0],[0,1,0],[self.dfdp
                (self.E_plus, self.xhat[0]), 0, 0]])
204         Ppred = F@self.P_state@F.T + self.Q_state
205         H = np.eye(3)
206         S = H@Ppred@H.T + self.Rs
207         K = (Ppred@H.T)@np.linalg.pinv(S)
208         self.xhat = xpred + K@(z - H@xpred)
209         self.P_state = (np.eye(3) - K@H)@Ppred
210
211         e = self.evaluate(xpred[0], z[2])
212         est_e = e[0]
213         var_e = e[1]
214
215         return self.xhat, est_e, var_e, self.P_state
```

Bibliography

[1] M. A. Said, R. N. Eppinga, E. Lipsic, N. Verweij, P. van der Harst, Relationship of arterial stiffness index and pulse pressure with cardiovascular disease and mortality, Journal of the American Heart Association 7 (2) (2018) e007621.

[2] J. A. Chirinos, P. Segers, T. Hughes, R. Townsend, Large-artery stiffness in health and disease: Jacc state-of-the-art review, Journal of the American College of Cardiology 74 (9) (2019) 1237–1263.

[3] C. Palombo, M. Kozakova, Arterial stiffness, atherosclerosis and cardiovascular risk: Pathophysiologic mechanisms and emerging clinical indications, Vascular pharmacology 77 (2016) 1–7.

[4] S. J. Zieman, V. Melenovsky, D. A. Kass, Mechanisms, pathophysiology, and therapy of arterial stiffness, Arteriosclerosis, thrombosis, and vascular biology 25 (5) (2005) 932–943.

[5] M. F. O'Rourke, M. E. Safar, Relationship between aortic stiffening and microvascular disease in brain and kidney: cause and logic of therapy, Hypertension 46 (1) (2005) 200–204.

[6] Y. Ben-Shlomo, M. Spears, C. Boustred, M. May, S. G. Anderson, E. J. Benjamin, P. Boutouyrie, J. Cameron, C.-H. Chen, J. K. Cruickshank, et al., Aortic pulse wave velocity improves cardiovascular event prediction: an individual participant meta-analysis of prospective observational data from 17,635 subjects, Journal of the American College of Cardiology 63 (7) (2014) 636–646.

[7] J. A. Chirinos, Arterial stiffness: basic concepts and measurement techniques, Journal of cardiovascular translational research 5 (2012) 243–255.

[8] E. Gkaliagkousi, S. Douma, The pathogenesis of arterial stiffness and its prognostic value in essential hypertension and cardiovascular diseases, Hippokratia 13 (2) (2009) 70.

[9] D. Nolte, C. Bertoglio, Inverse problems in blood flow modeling: A review, International Journal for Numerical Methods in Biomedical Engineering 38 (8) (2022) e3613.

[10] T. Namba, N. Masaki, B. Takase, T. Adachi, Arterial stiffness assessed by cardio-ankle vascular index, International journal of molecular sciences 20 (15) (2019) 3664.

[11] M. S. Taljanovic, L. H. Gimber, G. W. Becker, L. D. Latt, A. S. Klauser, D. M. Melville, L. Gao, R. S. Witte, Shear-wave elastography: basic physics and musculoskeletal applications, Radiographics 37 (3) (2017) 855–870.

[12] E. M. van Disseldorp, N. J. Petterson, F. N. van de Vosse, M. R. van Sambeek, R. G. Lopata, Quantification of aortic stiffness and wall stress in healthy volunteers and abdominal aortic aneurysm patients using time-resolved 3d ultrasound: a comparison study, European Heart Journal-Cardiovascular Imaging 20 (2) (2019) 185–191.

[13] v. D. Korteweg, Ueber die fortpflanzungsgeschwindigkeit des schalles in elastischen röhren, Annalen der Physik 241 (12) (1878) 525–542.

[14] A. Isebree Moens, Der erste wellengipfel in dem absteigenden schenkel der pulscurve, Archiv für die gesamte Physiologie des Menschen und der Tiere 20 (1) (1879) 517–533.

[15] L. M. Van Bortel, S. Laurent, P. Boutouyrie, P. Chowienczyk, J. Cruickshank, T. De Backer, J. Filipovsky, S. Huybrechts, F. U. Mattace-Raso, A. D. Protogerou, et al., Expert consensus document on the measurement of aortic stiffness in daily practice using carotid-femoral pulse wave velocity, Journal of hypertension 30 (3) (2012) 445–448.

[16] I. B. Wilkinson, C. M. McEniery, G. Schillaci, P. Boutouyrie, P. Segers, A. Donald, P. J. Chowienczyk, et al., Artery society guidelines for validation of non-invasive haemodynamic measurement devices: Part 1, arterial pulse wave velocity, Artery Research 4 (2) (2010) 34–40.

[17] J. J. O. A. Costa, R. d. C. A. Cunha, A. d. A. O. Alves Filho, L. R. Bessa, R. L. S. de Lima, A. dos Reis Silva, B. A. e Souza, C. de Almeida Viterbo,

M. B. Requião, D. Brustolim, et al., Analysis of vascular aging in arterial hypertension–population-based study: Preliminary results, Artery Research 25 (3-4) (2019) 131–138.

[18] C. Papadacci, T. Mirault, B. Dizier, M. Tanter, E. Messas, M. Pernot, Non-invasive evaluation of aortic stiffness dependence with aortic blood pressure and internal radius by shear wave elastography and ultrafast imaging, IRBM 39 (1) (2018) 9–17.

[19] J. Vappou, J. Luo, K. Okajima, M. Di Tullio, E. Konofagou, Aortic pulse wave velocity measured by pulse wave imaging (pwi): A comparison with applanation tonometry, Artery research 5 (2) (2011) 65–71.

[20] A. Sarvazyan, Diversity of biomedical applications of acoustic radiation force, Ultrasonics 50 (2) (2010) 230–234.

[21] V. Ayyalasomayajula, Ø. Ervik, H. Sorger, B. Skallerud, Macro-indentation testing of soft biological materials and assessment of hyperelastic material models from inverse finite element analysis, Journal of the Mechanical Behavior of Biomedical Materials (2024) 106389.

[22] L. Liang, M. Liu, J. Elefteriades, W. Sun, Pytorch-fea: Autograd-enabled finite element analysis methods with applications for biomechanical analysis of human aorta, Computer Methods and Programs in Biomedicine 238 (2023) 107616.

[23] B. Narayanan, M. L. Olender, D. Marlevi, E. R. Edelman, F. R. Nezami, An inverse method for mechanical characterization of heterogeneous diseased arteries using intravascular imaging, Scientific Reports 11 (1) (2021) 22540.

[24] C. Bertoglio, P. Moireau, J.-F. Gerbeau, Sequential parameter estimation for fluid–structure problems: application to hemodynamics, International Journal for Numerical Methods in Biomedical Engineering 28 (4) (2012) 434–455.

[25] A. Delalleau, G. Josse, J. Lagarde, H. Zahouani, J. Bergheau, Characterization of the mechanical properties of skin by inverse analysis combined with an extensometry test, Wear 264 (5-6) (2008) 405–410.

[26] M. Habibi, R. M. D'Souza, S. T. Dawson, A. Arzani, Integrating multi-fidelity blood flow data with reduced-order data assimilation, Computers in Biology and Medicine 135 (2021) 104566.

[27] A. Jonášová, J. Vimmr, Noninvasive assessment of carotid artery stenoses by the principle of multiscale modelling of non-newtonian blood flow in patient-specific models, Applied Mathematics and Computation 319 (2018) 598–616.

[28] D. Canuto, J. L. Pantoja, J. Han, E. P. Dutson, J. D. Eldredge, An ensemble kalman filter approach to parameter estimation for patient-specific cardiovascular flow modeling, Theoretical and Computational Fluid Dynamics 34 (2020) 521–544.

[29] A. Caiazzo, F. Caforio, G. Montecinos, L. O. Muller, P. J. Blanco, E. F. Toro, Assessment of reduced-order unscented kalman filter for parameter identification in 1-dimensional blood flow models using experimental data, International journal for numerical methods in biomedical engineering 33 (8) (2017) e2843.

[30] L. O. Müller, A. Caiazzo, P. J. Blanco, Reduced-order unscented kalman filter with observations in the frequency domain: application to computational hemodynamics, IEEE Transactions on Biomedical Engineering 66 (5) (2018) 1269–1276.

[31] D. Lombardi, Inverse problems in 1d hemodynamics on systemic networks: A sequential approach, International journal for numerical methods in biomedical engineering 30 (2) (2014) 160–179.

[32] K. Jain, S. Jain, A. Guha, A. Patra, An approach to early stage detection of atherosclerosis using arterial blood pressure measurements, Biomedical Signal Processing and Control 68 (2021) 102594.

[33] C. Bertoglio, D. Barber, N. Gaddum, I. Valverde, M. Rutten, P. Beerbaum, P. Moireau, R. Hose, J.-F. Gerbeau, Identification of artery wall stiffness: In vitro validation and in vivo results of a data assimilation procedure applied to a 3d fluid–structure interaction model, Journal of Biomechanics 47 (5) (2014) 1027–1034.

[34] D. Rüschen, M. Rimke, J. Gesenhues, S. Leonhardt, M. Walter, Online cardiac output estimation during transvalvular left ventricular assistance, Computer methods and programs in biomedicine 171 (2019) 87–97.

[35] D. A. Hullender, O. R. Brown, Simulations of blood pressure and identification of atrial fibrillation and arterial stiffness using an extended kalman

filter with oscillometric pulsation measurements, Computer Methods and Programs in Biomedicine 198 (2021) 105768.

[36] P. Shi, H. Liu, Stochastic finite element framework for simultaneous estimation of cardiac kinematic functions and material parameters, Medical Image Analysis 7 (4) (2003) 445–464.

[37] J. Jin, H. Zhang, X. Geng, Y. Zhang, T. Ye, The pulse waveform quantification method basing on contour and derivative, Computer Methods and Programs in Biomedicine 220 (2022) 106784.

[38] H. Zhang, B. Li, A. A. Young, P. J. Hunter, Recovery of myocardial kinematic function without the time history of external loads, EURASIP Journal on Advances in Signal Processing 2010 (2009) 1–9.

[39] Z. Gao, H. Xiong, X. Liu, H. Zhang, D. Ghista, W. Wu, S. Li, Robust estimation of carotid artery wall motion using the elasticity-based state-space approach, Medical image analysis 37 (2017) 1–21.

[40] W. Liu, S. Du, S. Zhou, T. Mei, Y. Zhang, G. Sun, S. Song, L. Xu, Y. Yao, S. E. Greenwald, Noninvasive estimation of aortic pressure waveform based on simplified kalman filter and dual peripheral artery pressure waveforms, Computer Methods and Programs in Biomedicine 219 (2022) 106760.

[41] C. Bertoglio, P. Moireau, J.-F. Gerbeau, Sequential parameter estimation for fluid–structure problems: application to hemodynamics, International Journal for Numerical Methods in Biomedical Engineering 28 (4) (2012) 434–455.

[42] J. Song, H. Xie, Y. Zhong, C. Gu, K.-S. Choi, Maximum likelihood-based extended kalman filter for soft tissue modelling, Journal of the Mechanical Behavior of Biomedical Materials 137 (2023) 105553.

[43] J. D. Humphrey, S. L. Delange, An introduction to biomechanics, Solids and Fluids, Analysis and Design. Springer, Heidelberg (2004).

[44] F. Irgens, Continuum mechanics, Springer Science & Business Media, 2008.

[45] G. A. Holzapfel, Nonlinear solid mechanics: a continuum approach for engineering science (2002).

[46] J. Bonet, R. D. Wood, Nonlinear continuum mechanics for finite element analysis, Cambridge university press, 1997.

[47] T. J. Pence, K. Gou, On compressible versions of the incompressible neo-hookean material, Mathematics and Mechanics of Solids 20 (2) (2015) 157–182.

[48] S. A. Maas, B. J. Ellis, G. A. Ateshian, J. A. Weiss, Febio: finite elements for biomechanics, Journal of biomechanical engineering (2012).

[49] V. C. Mow, S. Kuei, W. M. Lai, C. G. Armstrong, Biphasic creep and stress relaxation of articular cartilage in compression: theory and experiments, Journal of biomechanical engineering (1980) 73–84.

[50] E. Turco, Tools for the numerical solution of inverse problems in structural mechanics: review and research perspectives, European Journal of Environmental and Civil Engineering 21 (5) (2017) 509–554.

[51] R. C. Aster, B. Borchers, C. H. Thurber, Parameter estimation and inverse problems, Elsevier, 2018.

[52] S. Avril, S. Evans, et al., Material parameter identification and inverse problems in soft tissue biomechanics, Springer, 2017.

[53] F. Yaman, V. G. Yakhno, R. Potthast, A survey on inverse problems for applied sciences, Mathematical problems in engineering 2013 (1) (2013) 976837.

[54] J. Hadamard, Lecture on cauchy's problem in linear difierential equations, New York (1932).

[55] D. McLaughlin, L. R. Townley, A reassessment of the groundwater inverse problem, Water Resources Research 32 (5) (1996) 1131–1161.

[56] R. S. Reddy, D. Arepally, A. K. Datta, Inverse problems in food engineering: A review, Journal of Food Engineering 319 (2022) 110909.

[57] I. J. Myung, Tutorial on maximum likelihood estimation, Journal of mathematical Psychology 47 (1) (2003) 90–100.

[58] P. C. Hansen, Discrete inverse problems: insight and algorithms, SIAM, 2010.

[59] S. L. Brunton, J. N. Kutz, Data-driven science and engineering: Machine learning, dynamical systems, and control, Cambridge University Press, 2022.

[60] A. Chatterjee, An introduction to the proper orthogonal decomposition, Current science (2000) 808–817.

[61] N. G. Polson, V. Sokolov, Bayesian regularization: From tikhonov to horseshoe, Wiley Interdisciplinary Reviews: Computational Statistics 11 (4) (2019) e1463.

[62] P. Mc Carthy, Direct analytic model of the l-curve for tikhonov regularization parameter selection, Inverse problems 19 (3) (2003) 643.

[63] D. Calvetti, E. Somersalo, Inverse problems: From regularization to bayesian inference, Wiley Interdisciplinary Reviews: Computational Statistics 10 (3) (2018) e1427.

[64] D. Z. Huang, T. Schneider, A. M. Stuart, Iterated kalman methodology for inverse problems, Journal of Computational Physics 463 (2022) 111262.

[65] R. Gurajala, P. B. Choppala, J. S. Meka, P. D. Teal, Derivation of the kalman filter in a bayesian filtering perspective, in: 2021 2nd International Conference on Range Technology (ICORT), IEEE, 2021, pp. 1–5.

[66] M. S. Grewal, A. P. Andrews, Kalman filtering: Theory and Practice with MATLAB, John Wiley & Sons, 2014.

[67] M. Khodarahmi, V. Maihami, A review on kalman filter models, Archives of Computational Methods in Engineering 30 (1) (2023) 727–747.

[68] R. Furrer, T. Bengtsson, Estimation of high-dimensional prior and posterior covariance matrices in kalman filter variants, Journal of Multivariate Analysis 98 (2) (2007) 227–255.

[69] G. Bishop, G. Welch, et al., An introduction to the kalman filter, Proc of SIGGRAPH, Course 8 (27599-23175) (2001) 41.

[70] A. Attarian, J. J. Batzel, B. Matzuka, H. Tran, Application of the unscented kalman filtering to parameter estimation, Mathematical Modeling and Validation in Physiology: Applications to the Cardiovascular and Respiratory Systems (2013) 75–88.

[71] P. Shrivastava, T. K. Soon, M. Y. I. B. Idris, S. Mekhilef, Overview of model-based online state-of-charge estimation using kalman filter family for lithium-ion batteries, Renewable and Sustainable Energy Reviews 113 (2019) 109233.

[72] M. S. Islam, T. Lehtimäki, M. Juonala, M. Kähönen, N. Hutri-Kähönen, K. Kainulainen, H. Miettinen, L. Taittonen, K. Kontula, J. S. Viikari, et al., Polymorphism of the angiotensin-converting enzyme (ace) and angiotesinogen (agt) genes and their associations with blood pressure and carotid artery intima media thickness among healthy finnish young adults—the cardiovascular risk in young finns study, Atherosclerosis 188 (2) (2006) 316–322.

[73] B. Williams, G. Mancia, W. Spiering, E. Agabiti Rosei, M. Azizi, M. Burnier, D. L. Clement, A. Coca, G. De Simone, A. Dominiczak, et al., 2018 esc/esh guidelines for the management of arterial hypertension: The task force for the management of arterial hypertension of the european society of cardiology (esc) and the european society of hypertension (esh), European heart journal 39 (33) (2018) 3021–3104.

[74] Harvard apparatus, https://www.harvardapparatus.com/, accessed: 2022-10-17.

[75] Endress hauser, https://www.endress.com/en, accessed: 2022-10-17.

[76] Ctas manual, https://www.harvardapparatus.com/compact-transducer-amplifier.html#:~:text=to%20Request%20Quote-,The%20Compact%20Transducer%20Amplifier%20(CTA)%20is%20a%20DC%20bridge-,transducer%20must%20be%20ordered%20together. (March 2022).

[77] A. Sinek, M. Mesek, M. Rojczyk, J. Juszczyk, W. P. Adamczyk, J. Sturdy, B. Melka, A. Golda, M. Nowok, Z. Ostrowski, et al., Evaluating the precision and reproducibility of non-invasive deformation measurements in an arterial phantom, Measurement 216 (2023) 112904.

[78] Phantom, https://www.phantomhighspeed.com/, accessed: 2022-10-17.

[79] Irix, https://irixlens.com/, accessed: 2022-10-17.

[80] Ge ultrasound, https://gehealthcare-ultrasound.com/en/vivid-family/vivid-s70/, accessed: 2024-07-17.

[81] N. Otsu, Threshold selection method from gray-level histograms., IEEE Trans Syst Man Cybern SMC-9 (1979) 62–66. doi:10.1109/TSMC.1979.4310076.

[82] I. Sobel, An isotropic 3x3 image gradient operator, Presentation at Stanford A.I. Project 1968 (02 2014).

[83] R. O. Duda, P. E. Hart, Use of the hough transformation to detect lines and curves in pictures, Communications of the ACM 15 (1) (1972) 11–15.

[84] Atcor sphygmocor cp/cpv/cpvh, `https://www.medipment.pl/produkt/atcor-sphygmocor-cpcpvcpvh-87371`, accessed: 2024-07-22.

[85] Plotdigitizer: Version 3.1.5 (2024).
URL `https://plotdigitizer.com`

[86] T. O. Hodson, Root mean square error (rmse) or mean absolute error (mae): When to use them or not, Geoscientific Model Development Discussions 2022 (2022) 1–10.

[87] D. Simon, Optimal state estimation: Kalman, H infinity, and nonlinear approaches, John Wiley & Sons, 2006.

[88] Mts insight, `https://www.mts.com/en/products/materials/load-cells/universal-test-system-load-cells`, accessed: 2024-09-03.

[89] Q400-2d dantec dynamics, `https://www.limess.com/en/products/q400-digital-image-correlation`, accessed: 2024-09-03.

[90] S. Bastida, J. I. Eguiazábal, M. Gaztelumendi, J. Nazábal, On the thickness dependence of the modulus of elasticity of polymers, Polymer testing 17 (2) (1998) 139–145.

[91] M. Petrova, A. Gavino, Y. Li, C. S. McLachlan, Comparison of parameters for assessment of carotid stiffness and their association with carotid atherosclerosis in rural australian adults: A pilot study, Journal of Clinical Medicine 12 (8) (2023) 2935.

[92] S. Särkkä, Á. F. García-Fernández, Temporal parallelization of bayesian smoothers, IEEE Transactions on Automatic Control 66 (1) (2020) 299–306.

[93] S. Thrun, W. Burgard, D. Fox, Probalistic robotics, Kybernetes 35 (7/8) (2006) 1299–1300.

[94] C. K. Wikle, L. M. Berliner, A bayesian tutorial for data assimilation, Physica D: Nonlinear Phenomena 230 (1-2) (2007) 1–16.

[95] T. M. Donovan, R. M. Mickey, Bayesian Statistics for Beginners: a step-by-step approach, Oxford University Press, USA, 2019.

[96] M. I. Jordan, The conjugate prior for the normal distribution, Lecture notes on Stat260: Bayesian Modeling and Inference (2010).

[97] Z. Chen, et al., Bayesian filtering: From kalman filters to particle filters, and beyond, Statistics 182 (1) (2003) 1–69.

Summary

Keywords: cardiovascular, soft tissue, parameter estimation, stiffness estimation, Extended Kalman Filter, DEKF

Cardiovascular diseases are the major cause of death around the world. It is estimated that 20% of the population is affected by elevated arterial wall stiffness. Arterial stiffening is a pathophysiological marker for prediction and severity of cardiovascular diseases such as, myocardial infarction, heart failure, as well as stroke, dementia, atrial fibrillation, aneurysm rupture or renal disease.

The stiffness may be assessed by measuring Pulse Wave Velocity or Shear Wave Elastography. For the former method, length of the segment between two points of measurements is required, e.g. between femoral and carotid artery. This measurement is associated with large uncertainty and the resulting stiffness is an average value along the chosen segment. Shear Wave Elastography allows non-invasive and real-time measurement of the elasticity of soft tissues. Shear waves are generated by focused acoustic radiation force from a linear ultrasound probe. The velocity of the wave is related with Young's modulus of the investigated tissue.

The dissertation presents methodology for non-invasive arterial stiffness assessment by combining measurement of arterial wall displacement with ultrasound, applanation tonometry for pressure waveform measurement and solving inverse problem to estimate Young's modulus. The 3D Neo-Hookean and Linear Elastic model of the artery is solved by Finite Element Method open-source software FEBio.

For model validation and testing proposed methodology for inverse problem based on Dual Extended Kalman Filter, experiments on artificial artery with known mechanical properties were performed.

The proposed approach is then applied to medical data. The resulting estimation of Young's modulus falls into physiological range reported in literature.

Summary in Polish (Streszczenie)

Słowa kluczowe: układ krążenia, tkanka miękka, estymacja parametrów, estymacja sztywności, Rozszerzony Filtr Kalmana, DEKF

Choroby układu sercowo-naczyniowego są glówną przyczyną zgonów na całym świecie. Szacuje się, że 20% populacji cierpi na zwiększoną sztywność ściany tętnic. Sztywność tętnic jest indykatorem umożliwiającym predykcję i ocenę stopnia zaawansowania chorób układu naczyniowego, takich jak zawał, niewydolnośc serca, jak również udar mózgu, demencja, migotanie przedsionków, pęknięcie tętniaka czy choroby nerek.

Sztywność ściany tętnic może być zmierzona przez Pulse Wave Velocity bądź Shear Wave Elastography. Pierwsza metoda wymaga zmierzenia odcinka pomiędzy dwoma punktami pomiarowymi, na przykład między tętnicą udową i szyjną. Pomiar ten jest obarczony dużą niepewnością, a otrzymana wartość sztywności jest wartością uśrednioną dla całego odcinka. Shear Wave Elastography umożliwia bezinwazyjne badanie sztywności w czasie rzeczywistym. Fala (shear wave) jest generowana poprzez skupioną falę akustyczną emitowaną przez liniową głowicę USG. Prędkość rozchodzącej się fali w tkance jest skorelowana z modułem Younga badanej tkanki.

Rozprawa doktorska przedstawia metodologię bezinwazyjnego wyznaczania sztywności tętnic poprzez pomiar USG przemieszczania ściany tętnicy, pomiar przebiegu ciśnienia w tętnicy tonometrem aplanacyjnym wraz z rozwiązaniem problemu odwrotnego do wyznaczenia modułu Younga. Model 3D (Neo-Hookean i liniowe równanie konstytutywne) tętnicy jest rozwiązywane przy użyciu oprogramowania open-source FEBio przy użyciu metody elementów skończonych.

W celu walidacji modelu oraz przetestowaniu proponowanej metodologii dla

problemu odwrotnego opartej na podwójnym rozszerzonym filtrze Kalmana, przeprowadzono eksperymenty na sztucznej tętnicy o znanych właściwościch mechanicznych.

Proponowane podejście jest następnie stosowane do danych medycznych. Uzyskany moduł Younga mieści się w zakresie fizjologicznym podawanym w literaturze.