Faculdade de Engenharia da Universidade do Porto

Surrogate Modeling of Constitutive Models for Arterial Wall Behaviour

João Carlos Moutinho Gonçalves

Thesis work carried out in
KU Leuven Department of Mechanical Engineering
Soft Tissue Biomechanics Group

Master's in Mechanical Engineering
General Mechanical Engineering Branch

FEUP Supervisor: Prof. Dr. Marco Parente
KU Leuven Supervisors: Prof. Dr. Nele Famaey and Dr. Hadi Shirazi

07/2024

Resumo

O relatório seguinte foi criado no âmbito da unidade curricular de Dissertação, para obtenção do grau académico de Mestre em Engenharia Mecânica na FEUP (Faculdade de Engenharia da Universidade do Porto). Serve, assim, para documentar abordagens, metodologias, e conhecimentos de base e teóricos por de trás do projeto.

Surrogate modeling (modelação por substituição) é um método emergente em diferentes ramos da engenharia devido ao seu potencial na aproximação de sistemas complexos, especialmente quando não há possibilidade de avaliar estes sistemas com facilidade (quer seja computacionalmente ou experimentalmente). Em particular na medicina, determinar o momento apropriado para uma intervenção cirúrgica num paciente não é uma tarefa fácil, especialmente quando se fala de enfermidades arteriais, a intervenção nas quais depende de critérios voláteis. Adicionalmente, modelos biomecânicos não só são muitas vezes computacionalmente caros de executar, mas também a sua execução pode ser demasiado demorada e, por isso, desapropriada para situações de urgência. Modelos substitutos, quando treinados em dados fidedignos e perante, por exemplo, informação biométrica de um paciente, aparentam potencial para resolver as dificuldades descritas acima. Por isso, este projeto serve para aplicar este método de engenharia na replicação do comportamento arterial, através de dados extraídos de diferentes modelos constitutivos da biomecânica.

O presente relatório inicia-se com uma breve introdução que contextualiza a aplicabilidade de modelos de substituição na medicina, seguida por uma revisão bibliográfica de artérias de um ponto de vista anatómico, de mecânica dos meios contínuos, de modelação por substituição, e de quantificação de incerteza para modelos computacionais. Após esta revisão, a metodologia é apresentada, começando pela definição de modelos constitutivos para o comportamento arterial (WSE1, um modelo Neo-Hookeano, WSE2, um modelo GOH, e WSE3, um modelo que envolve Crescimento e Remodelação), seguido pela conceção de um desenho experimental para extrair dados dos modelos arteriais, e, por fim, o treino e avaliação de desempenho dos modelos substitutos. Observações acerca do trabalho desenvolvido e de eventuais aplicabilidades em situações de maior complexidade são providenciadas no final do relatório.

Em termos de notação matemática usada, símbolos em itálico e negrito representam vetores, símbolos em apenas negrito representam matrizes, e símbolos apenas em itálico representam escalares. Ao longo do relatório, as referências encontram-se posicionadas nos finais de frase e cumprem a formatação IEEE. O código desenvolvido encontra-se disponível em https://gitlab.com/joaojcmgsgroup/Thesis.

Abstract

The following report was created within the scope of the Dissertation course, to obtain the academic degree of Master in Mechanical Engineering at FEUP (Faculty of Engineering of the University of Porto). It serves to document the background, reasonings, methodology and theoretical knowledge associated with the project.

Surrogate modeling has become an increasingly important aspect in engineering fields for the approximation of complex systems, especially when these cannot be easily evaluated (be it computationally or be it experimentally). Particularly in the medical field, deciding when to perform a surgical intervention is not always an easy task, especially when it comes to arterial conditions, which depend on volatile criteria. Additionally, biomechanical mathematical models can be computationally costly to run, and take too long to output results for a tight time window. Surrogate models, when trained on reliable data and when given easily obtainable patient biomarkers as inputs, can be a potential solution for these problems. As such, this project serves to apply this engineering method in the depiction of arterial wall behaviour, with data extracted from different biomechanical constitutive models.

The report starts with a brief introduction regarding the context of surrogate modeling for medicine, followed by a literature review on arteries from an anatomical point of view, continuum mechanics, surrogate modeling and uncertainty quantification of computational models. After this, the methodology is presented, where the arterial constitutive models that we wanted to emulate via surrogates are defined (WSE1, a Neo-Hookean model, WSE2, a GOH model, and WSE3, a Growth and Remodeling type of model), followed by the conception of a design of experiments, the main aim of which was to extract data from the constitutive models, and finally, the training and posterior evaluation of the surrogates. A reflection on the difficulties of creating these metamodels and on their potential use in more elaborate scenarios is also provided in the end of the report.

Regarding mathematical notation conventions, vectors are represented by bold italic letters, matrices are represented by bold non-italic letters, and scalars are represented by italic non-bold letters. Throughout the report, references are located at the end of sentences and follow the IEEE style. All the code that was developed is freely available in https://gitlab.com/joaojcmgsgroup/Thesis.

Acknowledgements

I would firstly like to express my gratitude to my KU Leuven supervisors, Prof. Dr. Nele Famaey, and Dr. Hadi Shirazi, and also Dr. Lauranne Maes, for their openness and interest in hosting me in the Soft Tissue Biomechanics Group, as well for their guidance and words of advice throughout this project, which proved invaluable.

On the same note, I want to thank all the members of the KU Leuven Soft Tissue Biomechanics Group who so kindly received me in their office.

Back home, I want to thank my FEUP supervisor, Prof. Dr. Marco Parente, who also provided me with tools and resources for the development of the project.

Table of Contents

List of Figures

List of Tables

Acronyms and Symbols

List of Acronyms

AAA	Ascending Aortic Aneurysm
ADAM	Adaptive Moment Estimation
BNN	Bayesian Neural Network
CDF	Cumulative Distribution Function
CM	Constitutive Model
DNN	Deep Neural Network
DoE	Design of Experiments
DT	Digital Transformation
EIVS	Expanded Input Variable Space
FEM	Finite Element Method
FEUP	*Faculdade de Engenharia da Universidade do Porto*
GOH	Gasser-Ogden-Holzapfel
GPR	Gaussian Process Regression
G&R	Growth and Remodeling
HCMT	Homogenized Constrained Mixture Theory
HGO	Holzapfel-Gasser-Ogden
IVS	Input Variable Space
LHC	Latin Hypercube
MSE	Mean Square Error
NN	Neural Network
PDF	Probability Density Function
PINN	Physics-Informed Neural Network
RMSE	Root Mean Square Error
SEDF	Strain Energy Density Function
SM	Surrogate Model
UIVS	Unseen Input Variable Space
UMAT	User-defined Material
UQ	Uncertainty Quantification
WSE	Wall Stress Estimator

List of Symbols

$\mathbf{F}$	Deformation gradient
$\mathbf{F}_g$	Growth Deformation gradient
$\mathbf{F}_r$	Remodeling Deformation gradient
$\mathbf{C}$	Right Cauchy-Green deformation tensor
$\mathbf{b}$	Left Cauchy-Green deformation tensor
$\mathbf{E}$	Green-Lagrange strain tensor
$\mathbf{e}$	Euler-Almansi strain tensor
$\boldsymbol{\varepsilon}$	Logarithmic strain tensor
$\mathbf{S}$	Second Piola-Kirchoff stress tensor
$\mathbf{P}$	First Piola-Kirchoff stress tensor
$\boldsymbol{\sigma}$	Cauchy stress tensor
$\boldsymbol{x}$	Position in the current configuration
$\boldsymbol{X}$	Position in the reference configuration
$\boldsymbol{u}$	Displacement function
$\boldsymbol{M}_i$	Orientation vector of fiber family i
$\boldsymbol{b}$	Body force vector per unit mass
λ_{aa}	Stretch along principal direction a
λ_r^i	Remodeling stretch for the fiber constituent i
J	Volume change ratio
ν	Poisson's ratio
K	Bulk modulus
W	Strain energy density function
ρ^i	Mass density of constituent i
R^2	R-squared / Coefficient of Determination
lb_i	Lower boundary of variable i
ub_i	Upper boundary of variable i
elb_i	Expanded lower boundary of variable i
eub_i	Expanded upper boundary of variable i
ulb_i	Unseen lower boundary of variable i
uub_i	Unseen upper boundary of variable i
L	Length in the reference configuration
l	Length in the current configuration
R	Radius in the reference configuration
r	Radius in the current configuration

1. Introduction

In today's world, an increasing number of industries seek to embrace technology with the purpose of improving current processes, bringing about innovation and increasing competition in their respective sectors.

This process is called Digital Transformation (DT) and can be generally described as an organizational transfiguration through the implementation of digital technologies [1]. In service-type industries such as entertainment, healthcare, education or transportation, DT serves as an approach to improve the quality that is delivered by a given service, benefiting the service providers by boosting agility, flexibility and improved organizational performance, and also empowering the consumer by fulfilling their demands and enhancing their experience with said service [1, 2]. Exemplary implementations of DT, across different industries, are online banking, telemedicine, patient-specific diagnosing, subscription-based platforms, mobile applications, etc. [2].

Delving deeper into DT for healthcare, it specifically aims to improve healthcare delivery and to provide solutions for medical problems in better and more innovative ways. According to [3], papers in this area of research tend to be focused on the following aspects:

- information technology (employment of real-life data to create, for example, patient specific medicine)
- e-health (broad term for improved digital medical technology)
- telemedicine (improved accessibility of healthcare services and providers)
- security in e-health (technologies to ensure sensitive medical information is preserved when stored in the cloud)
- education in e-health (improving transmission of knowledge in medicine)

The scope of this project is related to the first of these five topics: by leveraging *in silico* (computational) modeling and the creation of digital representation of real life systems (such as clinical specimens, or human subsystems), it is possible to contribute to improved medical diagnosis and the development of innovative medical solutions (by means of new drugs, surgery techniques, etc.) [4, 5].

However, these high-fidelity virtual representations of large, multiscale systems often involve highly complex differential equations, resulting in high computational resources and high processing times which deter from performing different computational methods such as sensitivity analysis, optimizations or uncertainty quantification. As such, surrogates (or metamodels) surge as a way to bypass these difficulties by mapping the relation between inputs and outputs provided by the digital twin itself, providing a tradeoff between accuracy and speed. This results in a model which acts as a supplementary tool for the computational analysis of the digital twin in question [6, 7].

Given this, this project was developed with the potential use of surrogates for biomedical applications in mind, particularly, for arterial modeling, given the uncertain nature of the illnesses that afflict these structures, requiring well-informed decision-making on how/when to perform high risk interventions to improve the patient's health. Such is the case with, for example, aneurysms, which, not only present high mortality rates, but whose intervention criteria aren't infallible in predicting when rupture occurs, leading researchers to new approaches to decision-making in aneurysm management [8].

In this project specifically, the aim was to employ the potential capabilities of surrogate modeling in the approximation of arterial wall constitutive models, and their uncertainty analysis. These constitutive models, though not as elaborate as digital twins, still possess enough complexity to challenge the modeling capabilities of surrogates.

Through this work, the goal is not only to develop technical skills in computational modeling for biomechanics, but also to gain insights into the critical steps involved in such modeling processes. This includes understanding how to develop continuum mechanics models, handle data, implement effective computational techniques, and address the challenges associated with uncertainty quantification in biomechanical simulations.

2. Literature Review

In the following section, a revision of the theoretical background is provided to the reader for a better understanding of the project at hand. As such, it is possible to define the following subsections:

- 1. Medical Background on Arteries
- 2. Constitutive Modeling
- 3. Surrogate Modeling
- 4. Uncertainty Quantification

Each one starts with a contextualization that serves to justify its insertion within the project at hand. Followed by this are the due specifications and detailing of the subtheme.

In certain instances, it is assumed that the reader is knowledgeable in what is being addressed, and not much background specifications may be given (for example, given the placement of this report within engineering studies, some aspects of continuum mechanics are assumed to be known by the reader).

2.1. Medical Background on Arteries

Blood vessels serve as important structures in the human body given their role of transporting blood to and from the cells in the body in order to ensure that all the necessary biophysical exchanges are executed (such as giving oxygen and nutrients to cells and receiving from them carbon dioxide). These channels form a closed circuit (the circulatory system), at the center of which the heart is located, responsible for cyclically pumping the blood through these channels [9, 10].

In terms of categorization, there are three types of blood vessels: arteries (carry blood away from the heart), arterioles (ramifications of arteries), veins (carry blood toward the heart), venules (ramifications of veins) and capillaries (thin walled vessels that act as a bridge between venules and arterioles and allow for oxygen/nutrients to be traded with carbon dioxide/waste products) [10, 11].

Due to their critical importance in maintaining good health, conditions and disorders associated with the vascular system are seen as major concerns that require effective treatment in order to prevent further complications that may put the life of a patient at risk [9]. Common conditions of this nature include aneurysms (bulging in weak/damaged areas of the vessel which may lead to rupture), atherosclerosis (buildup of plaque which is responsible for lack of blood flow in critical organs such as the heart or brain), vasculitis (inflammation of the blood vessel and subsequent narrowing of the flow), etc. [9, 10].

When delving into these illnesses in more detail, it is possible to highlight one in particular: the rupture of an AAA (ascending aortic aneurysm). By carrying a mortality rate of 80-90% and since 50% of patients die before reaching the hospital, AAA garners reputation of being one of the most life-threatening arterial conditions and the cause of about 4-5% of sudden deaths [12-14]. This is due to the silent nature of the condition, since its development goes unnoticed for a long time, and also due to the proximity of the aorta to the heart, leading to mass internal bleeding when the rupture occurs [14].

As such, to better understand this condition and how it occurs, it is important to have a basic knowledge on the medical/anatomical concepts that characterize arteries, which is the purpose of the following subsections.

2.1.1. Constituent Tissues

Arteries are made up of different types of tissues which grant them properties to perform different functions. In animals, soft tissues can be grouped as either epithelial, muscle, nervous or connective tissue. Arteries are mostly made up of muscle (smooth muscle cells) and connective tissues (collagen and elastin) [15].

2.1.1.1. Smooth Muscle

In the case of smooth muscle, it plays a vital role in blood pressure regulation and tissue oxygenation. More specifically, its primary function is in contraction based on the presence of calcium (higher presence of this ion leads to higher strength in contraction), which allows for the regulation of the blood flow and pressure via vascular resistance. Additionally, it is also present in multiple other soft tissues (in the gastrointestinal tract, the bladder, the respiratory tract, etc.) where it plays just as important roles. If one delves deeper on the issue, it is possible to differentiate between single-unit (allows for synchronous contraction of the cells based on a singular stimulus from a nerve) and multi-unit smooth muscle (each muscle cell activated by a different nerve, allowing finer control), the latter of which is the one that is prevalent in arteries [16].

2.1.1.2. Connective Tissue

On the other hand, connective tissue can be divided into two components: cells and the extracellular matrix. The former consists of macrophages, leukocytes, etc., while the latter is made of a ground substance (gelatinous material that acts as a glue between the matrix and the cells and allows for nutrient exchange between cells and blood vessels), and of fibers, which can be of two types in arteries:

- Collagenous fiber (collagen): large, flexible and strong fibers that provide resistance to tension and strain to the matrix.
- Elastic fibers (elastin): thin fibers that allow for stretching and recoil (elasticity) in the artery.

Just as the smooth muscle, connective tissue is present all over the body, throughout all organs [15].

2.1.2. Structure

Arteries, as previously said, carry blood away from the heart to the rest of the body. Two arteries branch off from the heart: the pulmonary artery and the aorta. The latter is the largest artery of the body, carrying large amounts of oxygenated blood to the cells through its ramifications (arterioles and capillaries) [17]. In terms of structure, arteries have three main layers, as suggested by Figure 1.

2.1.2.1. Tunica Intima

The innermost layer, the tunica intima, interacts with the contents of the lumen (inside volume of the artery), and it is made up of endothelial cells (thin layer that regulates exchanges between blood and surrounding tissues) and a supporting layer made up of muscle, collagen and elastic fibers. The boundary that separates the tunica intima from the tunica media is named tunica interna and is mostly elastic [9].

2.1.2.2. Tunica Media

The following layer, the tunica media, is significantly thicker than the surrounding ones. Due to its composition being mostly smooth muscle fibers (orientated mostly circumferentially), it can change the internal diameter of the vessel through dilations or constrictions, and, subsequently, the pressure in the interior (higher diameter leads to lower pressure, and vice versa) [9, 18, 19].

2.1.2.3. Tunica Adventitia

Finally, the tunica adventitia is the outermost layer and is responsible for the integrity of the vessel and its ability to sustain wall strains, and is made up primarily of elastic and collagenous fibers [9].

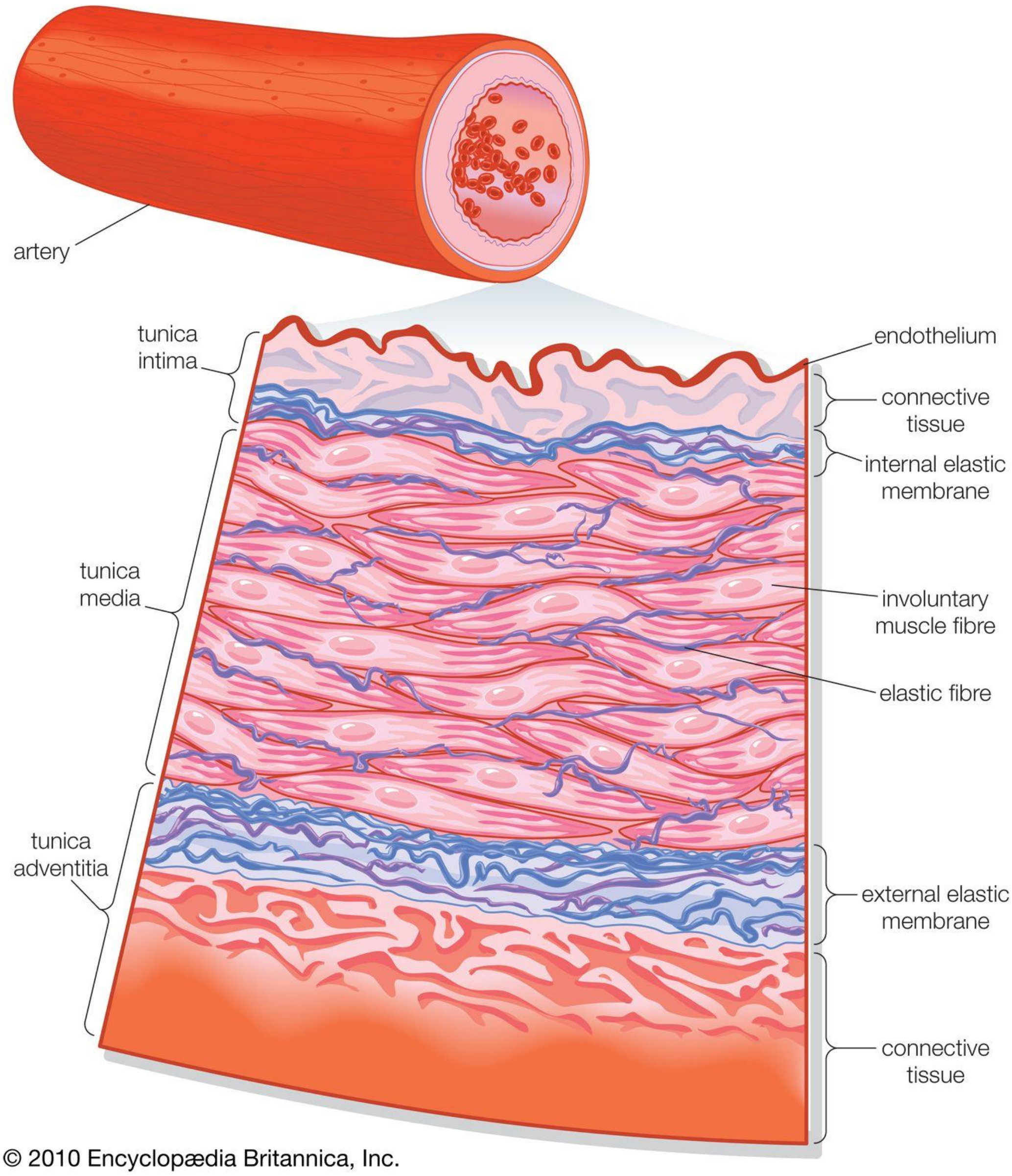

Figure 1 - Transverse section of an artery [17]

It should be noted that slight variations in the layers occur depending on the type of artery. Larger arteries (such as the aorta), also called elastic arteries, possess thicker tunicae intimae, thinner tunicae adventitiae, larger diameters and are closer to the heart, while medium sized arteries (or muscular arteries) are in the periphery and possess high density of smooth muscle in the tunica media [9, 20].

2.1.3. Material Behaviour

When describing arteries under a biomechanical lens, it is possible to identify a set of characteristics and phenomena that make up their behaviour as a biological material.

The first one, which has been previously mentioned, is its nonlinear elasticity (hyperelasticity). The ability to be loaded and, when unloaded, return to an undeformed shape is granted by the presence of elastin and collagen fibers. More specifically, there is a consensus that arteries possess cylindrical orthotropy, i.e. its stress-strain response varies depending on whether it is being loaded radially, circumferentially or axially [20, 21].

Though the elastic aspect is typically the one that is mostly focused on when characterizing their material behaviour, arteries also possess inelastic responses that are typically neglected in studies, as they complicate significantly problem formulation [20]. One of those inelastic components is viscoelasticity. This means that the stress-strain relation has a dependency on time which can be of three types: creep (variable strain in time under constant stress), relaxation (variable stress in time under constant strain) and hysteresis (differences in loading and unloading under a cyclical pattern). This last aspect is particularly interesting when we consider that arteries are cyclically loaded in real-time due to the intraluminal pressure that blood exerts on the inner walls wall during heart pumping cycles; Figure 2 illustrates the pressure-time (stress) and diameter-time (strain) relations during one heart cycle [20, 22]. It is believed that smooth muscle is the main responsible for this type of response and, thus, more noticeable in medium-sized arteries [19].

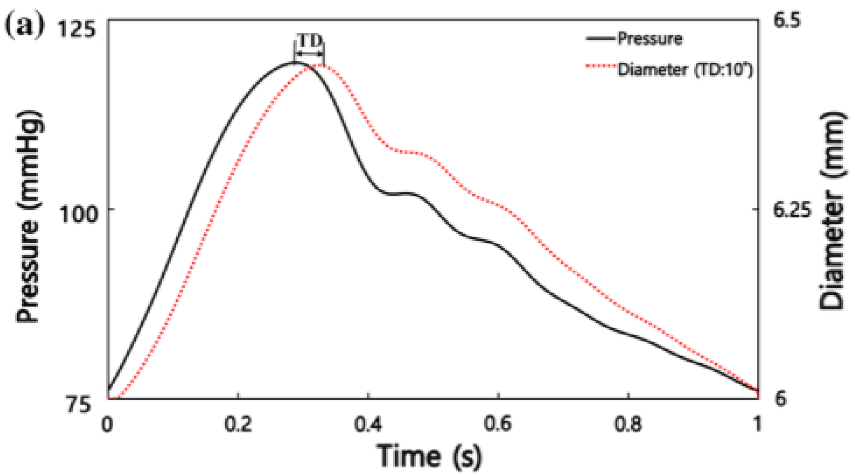

Figure 2 - Typical pressure and diameter waveforms for a time delay TD for a medium-sized artery [22]

An additional inelastic feature has to do with the ability of remodeling that the constituents of arteries have. In other words, arterial remodeling refers to functional and structural changes in their microstructure due to interactions with the surrounding environment and other biological factors (namely, disease, injury, ageing, hypertension, inflammation, etc.) that cause a change in the lumen dimensions of a vessel, compared to standard conditions. The remodeling of the artery can be inward (decrease of the lumen) or outward (increase of lumen), and can also be hypotrophic, eutrophic or hypertrophic when accompanied by decreasing, constant or increasing wall thickness, respectively (Figure 3). This phenomenon occurs mostly in larger arteries, especially the inwardly eutrophic or hypertrophic types [23].

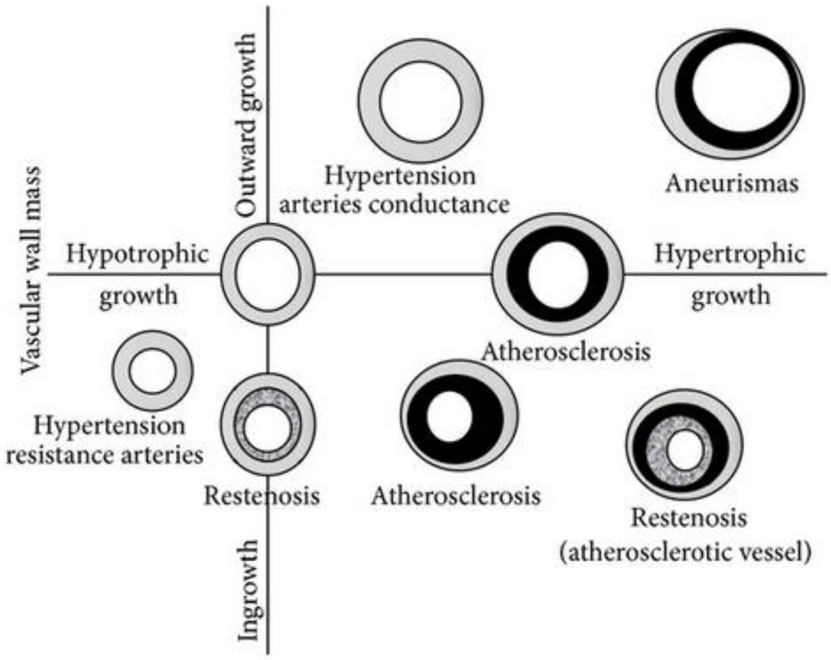

Figure 3 - Schematic representation for the adaptation of these changes in different pathologies [24]

The remodeling of arterial tissue leads to residual stresses, meaning that, if an arterial transverse arterial segment were cut radially, it would recoil at an angle. The residual stresses serve to lower transmural gradients, i.e. it acts as a means to uniformize circumferential strain and stress along the wall thickness for a mean arterial pressure [19, 23].

Another aspect of arteries has to do with their water content. The composite-like structure of elastin, smooth muscle and collagen, along with the high water contents typical of soft tissues (70-80% water) justify formulations that consider their full incompressibility (resistance to volume change when loaded) [19, 20, 25].

2.1.4. Relevant Illnesses

2.1.4.1. Atherosclerosis

When it comes medical conditions that affect arteries, one should mention atherosclerosis. It consists in the buildup of plaque (specifically, lipoproteins) inside the tunica intima. The stages of development of atherosclerosis are [26]:

- Endothelial damage: inflammation is generated as an immune response to damage.
- Fatty streak formation: first visible sign of atherosclerosis, consisting of dead cells in the endothelium.
- Plaque accumulation: when the plaque grows sufficiently, fibrous layers form around it to avoid bits from breaking off into the bloodstream.
- Plaque rupture: the fibrous layer breaks, generating blood clots which obstruct the lumen even more

The main complications that arise from this illness are related to the occlusion of the lumen, affecting normal blood flow and the oxygenation of bodily tissues, and blood clotting that breaks off and ends up obstructing other parts of the body, such as coronary (heart) arteries or brain arteries [26]. Mechanically, the accumulation of plaque generates a mechanical stimulus for remodeling of the vessel structure, as Figure 3 suggests (hypertrophic remodeling).

The most common risk factors include genetics, age, sex, and lifestyle factors (such as smoking, alcoholism, diet, etc.). In terms of mortality rates, it is the underlying cause of about 50% of all deaths in western society, and, yearly, 1 in every 4 deaths in the United States are due to heart disease (mediated by atherosclerosis). Its asymptomatic development, allied with lifestyle habits, is one of the reasons for the big impact it has [26].

2.1.4.2. Aneurysms

Aneurysms are another type of condition that commonly affects arteries, being defined as permanent localized dilatations of the vessel diameter by at least 50%. When occurring in the aorta, life threatening situations arise and, when diagnosed, require prompt treatment [27].

The development of the illness can be explained by the fragmentation of elastin and smooth muscle fibers and replacement by amorphous material, which occurs with age, and leads to increased stiffness and weakening of the vessel wall. Furthermore, mechanically, the dilation leads to higher wall stresses, inducing arterial remodeling (specifically, hypotrophic remodeling of the vessel, Figure 3) which causes even more aortic dilation [28]. Once the peak wall stress exceeds the local strength, rupture followed by massive bleeding occurs.

The most common risk factors are similar to those of atherosclerosis and atherosclerosis itself can also cause aneurysms. Once again, due to its silent nature, but also because of the severity of bleeding after rupture, mortality reaches 80% in some studies [27].

When a patient has been diagnosed in a timely manner with AAA, there are two used criteria that are taken into account when deciding surgical intervention: a geometric criterion that indicates a danger zone when the dilation diameter is higher than 55m, and a growth criterion that advises surgical intervention if the rate of increase of the dilation diameter exceeds 1 cm/year, approximately. However, ruptures don't always occur when these criteria indicate so, leading to their low reliability. Other criteria have been suggested, such as using the peak wall stress; however, difficulties in obtaining the peak wall stress (either through *in vivo* measuring or computational simulations) and the fact that wall strength is patient-specific don't facilitate the usage of this method [8].

2.2. Constitutive Modeling

Constitutive modeling is essential in problem formulation for biomechanics, since it allows for the mathematical representation and recreation of real-life mechanical behaviours. Choosing a particular constitutive framework serves to reflect specific behaviours of the tissue at hand. This, however, involves decision making in terms of the complexity/realism that one wants to implement into the model and, thus, one should always weigh the trade-offs of having more intricate but computationally heavy models, or simple but not so very realistic models.

This subsection on constitutive modeling starts with equations from a continuum mechanics background, and, subsequently, relevant constitutive models are explained. Next, equations that explain the kinematics of hollow tubes are listed and finally, a brief overview of the finite element method and on Abaqus UMAT building is given to the reader.

2.2.1. *Continuum Mechanics Principles*

Stress-strain makes up the most important relation in continuum mechanics and, therefore, in constitutive modeling. The presented equations are base expressions for finite deformations, commonly enunciated in all types of reports in the biomechanics field and can be found in more details in references such as [29, 30].

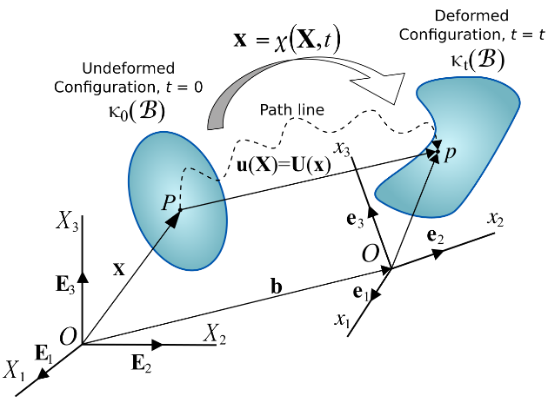

Figure 4 - Schematic that represents the state of finite deformation [29]

In finite deformation, one must start by defining the deformation gradient, $\mathbf{F}$:

$$\boldsymbol{x} = \Phi(\boldsymbol{X}, t) = (\boldsymbol{X}, t) + \boldsymbol{X} \qquad (1)$$

$$\mathbf{F} = \boldsymbol{x} \otimes \boldsymbol{\nabla} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} \qquad (2)$$

where $\boldsymbol{x}$ is the position vector in the deformed configuration, $\boldsymbol{X}$ is the deformation vector in the undeformed configuration, t is the time variable and $\boldsymbol{\nabla}$ is the Nabla operator.

Deformation gradients, in many biomechanical problems, are of the following type:

$$\mathbf{F} = \begin{bmatrix} \lambda_{11} & 0 & 0 \\ 0 & \lambda_{22} & 0 \\ 0 & 0 & \lambda_{33} \end{bmatrix} \tag{3}$$

where λ_{ii} is the stretch in direction ii.

If the coordinate system used is rectangular, the subscripts of the stretches in (3) are actually $(1,2,3) = (x, y, z)$. On the other hand, if the coordinate system is cylindrical, the subscripts are $(1,2,3) = (r, \theta, z)$.

The importance of the deformation gradient lies in its mathematical representation of loading scenarios. Some exemplary gradients used in biomechanical applications are:

Uniaxial loading with stretch λ along direction 1, with volume preservation, and isotropic behaviour	$\lambda_{11} = \lambda$ $\lambda_{22} = \frac{1}{\sqrt{\lambda}}$ $\lambda_{33} = \frac{1}{\sqrt{\lambda}}$	(4)
Biaxial loading with stretch λ along directions 1 and 2, with volume preservation and isotropic behaviour	$\lambda_{11} = \lambda$ $\lambda_{22} = \lambda$ $\lambda_{33} = \frac{1}{\lambda^2}$	(5)
Biaxial loading with stretch λ along direction 1 and direction 2 is constrained, with volume preservation and isotropic behaviour	$\lambda_{11} = \lambda$ $\lambda_{22} = 1$ $\lambda_{33} = \frac{1}{\lambda}$	(6)

with volume preservation being ensured when $\det(\mathbf{F}) = J = \lambda_{11}\lambda_{22}\lambda_{33} = 1$.

After having defined our deformation gradient, it is necessary to define the deformation tensors. However, it is necessary to decide whether the model being built should follow a fully incompressible (zero volume change) or a near incompressible approach (near zero volume change). In case of the former, Poisson's ratio becomes $\nu = 0.5$, leading to an infinite bulk modulus K and an increased stiffness and complexity of the model which is added at the end. In case of the latter, the deformation tensors used in the computations need to be their isochoric components (deformations without changes in volume, represented with a slash above the tensor in mathematical notation) and a volumetric component needs to be added to the strain energy density function in order to enforce incompressibility [31].

Mathematically speaking, this is translated as:

$$\mathbf{C} = \mathbf{F}^{\mathrm{T}}\mathbf{F} \ / \ \bar{\mathbf{C}} = J^{-2/3}\mathbf{C} \quad (7)$$
$$\mathbf{b} = \mathbf{F}\mathbf{F}^{\mathrm{T}} \ / \ \bar{\mathbf{b}} = J^{-2/3}\mathbf{b} \quad (8)$$

where $\mathbf{C}$ and $\mathbf{b}$ are the Right and Left Cauchy-Green deformation tensors, respectively, and $\bar{\mathbf{C}}$ and $\bar{\mathbf{b}}$ their deviatoric components. Their respective invariants (I_i and $\bar{I}_i$) are defined as:

$$I_1 = \mathrm{tr}(\mathbf{C}) \ / \ \bar{I}_1 = \mathrm{tr}(\bar{\mathbf{C}}) \quad (9)$$
$$I_2 = \frac{1}{2}[(\mathrm{tr}(\mathbf{C}))^2 + \mathrm{tr}(\mathbf{C}^2)] \ / \ \bar{I}_2 = \frac{1}{2}[(\mathrm{tr}(\bar{\mathbf{C}}))^2 + \mathrm{tr}(\bar{\mathbf{C}}^2)] \quad (10)$$
$$I_3 = \det(\mathbf{C}) = J^2 = 1 \ / \ \bar{I}_3 = \det(\bar{\mathbf{C}}) = \left(J^{-2/3}\right)^2 \quad (11)$$

and the subsequent strain tensors,

$$\mathbf{E} = \frac{1}{2}(\mathbf{C} - \mathbf{I}) \ / \ \bar{\mathbf{E}} = \frac{1}{2}(\bar{\mathbf{C}} - \mathbf{I}) \quad (12)$$
$$\mathbf{e} = \frac{1}{2}(\mathbf{I} - \mathbf{b}^{-1}) \ / \ \bar{\mathbf{e}} = \frac{1}{2}\left(\mathbf{I} - \bar{\mathbf{b}}^{-1}\right) \quad (13)$$
$$\boldsymbol{\varepsilon} = \ln\left(\mathbf{C}^{1/2}\right) \ / \ \bar{\boldsymbol{\varepsilon}} = \ln\left(\bar{\mathbf{C}}^{1/2}\right) \quad (14)$$

where $\mathbf{E}$, $\mathbf{e}$ and $\boldsymbol{\varepsilon}$ are the Green-Lagrange, Euler-Almansi and Logarithmic strain tensors, respectively, and $\bar{\mathbf{E}}$, $\bar{\mathbf{e}}$ and $\bar{\boldsymbol{\varepsilon}}$ their deviatoric components.

The next step is to compute the stress tensor. To do this, a strain energy density function (SEDF) W is defined in function of the deformation gradient (and its stretches) or in function of a deformation tensor (and its invariants), and, by differentiating the SEDF in reference to said tensor, different stress tensors are obtained. The conjugate stress-strain pairs are:

$$\mathbf{S} = \frac{\partial \mathrm{W}}{\partial \mathbf{E}} \quad (15)$$

$$\mathbf{P} = \frac{\partial \mathrm{W}}{\partial \mathbf{F}} \quad (16)$$

$$\boldsymbol{\sigma} = \frac{\partial \mathrm{W}}{\partial \boldsymbol{\varepsilon}} \quad (17)$$

where $\mathbf{S}$, $\mathbf{P}$ and $\boldsymbol{\sigma}$ are the Second Piola-Kirchhoff, First Piola-Kirchhoff and Cauchy stress tensors, respectively. The relations between the stress tensor are:

$$\mathbf{P} = \mathbf{F}\mathbf{S} \quad (18)$$
$$\boldsymbol{\sigma} = \frac{1}{J}\mathbf{F}\mathbf{S}\mathbf{F}^{\mathrm{T}} \quad (19)$$

In the case of incompressibility, an extra component gets added to the stress tensor, $-p\mathbf{I}$, where p is a Lagrange Multiplier which needs to be calculated based on the boundary conditions of the problem. For near incompressibility, a volumetric component, W_{vol}, gets added to the SEDF, which depends on the volume change ratio J, and only after does it go through differentiation.

2.2.2. Material Models

Material models serve to define the previously mentioned SEDF, allowing the computation of the stress tensor. To put it in another way, it is an expression that relates the stress response of tissue under specific strain conditions, which is dependent on the relevant material properties.

There are many different material models that are meant to capture specific behaviours of soft tissues. The ones that are described below are the ones which were relevant for the project at hand.

2.2.2.1. Neo-Hookean (Hyperelasticity)

The Neo-Hookean material model is a very simple hyperelastic material model which is widely used in behavioural studies of rubber-like materials. It requires only one material parameter (C_{10}) and is defined as [31]:

$$W = C_{10}(I_1 - 3) \tag{20}$$

2.2.2.2. Gasser-Ogden-Holzapfel (Anisotropy)

The Gasser-Ogden-Holzapfel (GOH) material model was proposed by T. C. Gasser, R. W. Ogden and G. A. Holzapfel in 2005 and serves as a generalization of the Holzapfel-Gasser-Ogden (HGO) model (proposed in 2000, [32]), and is described in detail in reference [20]. This model seeks to capture the hyperelastic isotropic behaviour of an elastin matrix and the anisotropic behaviour of dispersed collagen fibers within arterial layers. This is done by introducing an extra component to the Neo-Hookean model, which depends on pseudo-invariants of the deformation tensors and captures the influence of fiber families in the tissue. The difference from the HGO model lies in the presence of the κ material parameter, which characterizes the dispersion of the fiber families. It is defined as:

$$W = W_{iso} + W_{anis} = C_{10}(I_1 - 3) + \frac{k_1}{2k_2}\sum_i [\exp(k_2\langle E_i\rangle^2) - 1] \tag{21}$$

$$E_i = \kappa I_1 + (1 - 3\kappa)I_i - 1 \tag{22}$$

$$I_i = \mathbf{C} : (\boldsymbol{M}_i \otimes \boldsymbol{M}_i) \tag{23}$$

where C_{10}, k_1, k_2 and κ are material parameters, M_i is the fiber orientation of the i-th family and I_i is the pseudo-invariant.

The most exemplary use of this model involves two collagen fiber families (designated by $i = 4$ and $i = 6$), the orientation of which is defined by the orientation vectors $\boldsymbol{M}_4 = [0 \quad \cos(\alpha) \quad \sin(\alpha)]'$ and $\boldsymbol{M}_6 = [0 \quad \cos(\alpha) \quad \sin(-\alpha)]'$, where α is the slope of the orientation of one of the families. For the case of arteries, this fiber structure provides them with anisotropic (more specifically, transversely isotropic) behaviour in the $\theta - z$ plane (normal to the radial direction r) (represented in Figure 5).

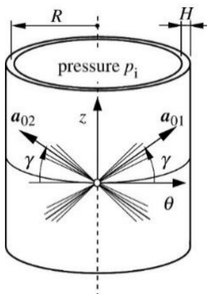

Figure 5 - Visual representation of the orientation of the fiber families on a hollow cylinder ($\boldsymbol{a}_i$ are the orientation vectors and γ is the slope angle) [20]

2.2.2.3. Homogenized Constrained Mixture Theory (Growth and Remodeling)

The Homogenized Constrained Mixture Theory (HCMT), proposed in 2016 by C. J. Cyron, R. C. Aydin and J. D. Humphrey (reference [33]), is a more computationally efficient adaptation of the Constrained Mixture Theory, proposed in 2002 by J. D. Humphrey and K. R. Rajagopal (reference [34]), both used to model growth and remodeling (G&R) behaviour of soft tissues. In this subsection, the equations that define the model are listed and briefly explained, but for a more comprehensive understanding, the reader is redirected to references [33-37].

At its core, this framework to mathematically depict growth (changes in mass) and remodeling (changes in structure) of arteries involves tracking creation and degradation of constituents by means of a series of mass balance equations and deformation gradients over a period of time [34].

To explain in other words, one may define the reference configuration as a mixture of arterial constituents (such as collagen) which suffers a process of continuous mass deposition and degradation called turnover, and which is also subjected to different boundary conditions. The process of constituent turnover leads to remodeling due to the difference in stress states between the newly deposited and the extant constituents. The homogenized aspect of the model comes from the fact that the model captures the stress-free configurations of the mass increments in a temporally homogenized way by using an adaptation of the extant stress-free configuration of the constituent, avoiding the more computationally intensive task of tracking these configurations for each new cohort of constituent that joins the mixture (as is depicted in Figure 6) [37].

The growth component of the model refers to a type of inelastic deformation that causes volume change due to mass increase or decrease. This is once again achieved by usage of a growth deformation gradient that is dependent on the time and on the constituent it applies to.

One aspect that should be highlighted, however, is that this model was developed for specific mixtures where remodeling occurs only in unidimensional constituents (anisotropic fibers) that are in mixture with a three-dimensional constituent (isotropic matrix). Although possible to depict remodeling of the matrix, the computations are not as direct as the fiber remodeling case [35, 38].

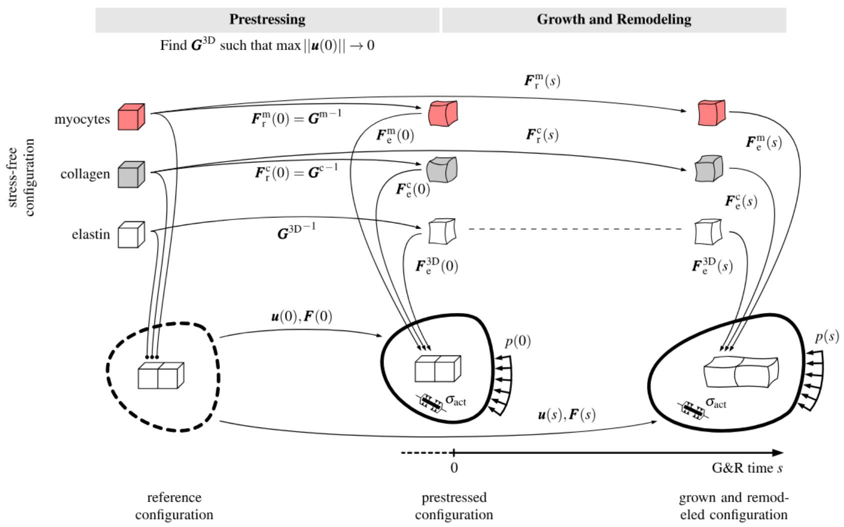

Figure 6 - Schematic for the HCMT material model [37]

Given this brief clarification, one may start by listing the mass balance equations:

$$\rho^{tot}(s) = \sum_i \rho^i(s) \qquad (24)$$

$$\phi^i(s) = \frac{\rho^i(s)}{\rho^{tot}(s)} \qquad (25)$$

where ρ^i is the mass density of constituent i, ϕ^i is the mass fraction of constituent i, and ρ^{tot} is the total mass density of the mixture, and all the variables are in function of the time variable s.

The stress state of the mixture, $\boldsymbol{\sigma}^{tot}(s)$, is defined as:

$$\boldsymbol{\sigma}^{tot}(s) = \sum_i \phi^i(s)\boldsymbol{\sigma}^i(s) \qquad (26)$$

where $\boldsymbol{\sigma}^i$ is the Cauchy stress tensor of constituent i. The computation of $\boldsymbol{\sigma}^i$ involves the typical algorithm that has been previously mentioned.

For the matrix, an isotropic material model is chosen and then differentiated in reference to the deformation gradient, while, for the fibers, the chosen material model is anisotropic. The difference here lies in the deformation gradient that is used to differentiate in reference to; that is, the gradient in question is the elastic deformation gradient of constituent i, $\mathbf{F}_e^i$, defined as:

$$\mathbf{F}_e^i(s) = \mathbf{F}(s)\mathbf{F}_g(s)^{-1}\mathbf{F}_r^i(s)^{-1} \tag{27}$$

where $\mathbf{F}_g$ is the growth deformation gradient, $\mathbf{F}_r^i$ is the remodeling deformation gradient for constituent i, and $\mathbf{F}$ is the deformation gradient for the whole mixture. The first of these gradients is dependent on the growth direction and the constituent mass densities, being defined as:

$$\mathbf{F}_g(s) = \frac{\rho^{tot}(s)}{\rho^{tot}(0)}\boldsymbol{a}_g \otimes \boldsymbol{a}_g + \left(\mathbf{I} - \boldsymbol{a}_g \otimes \boldsymbol{a}_g\right) \tag{28}$$

where $\boldsymbol{a}_g$ is the preferred direction of growth. Notice that for the specific case of $\boldsymbol{a}_g = [1 \quad 1 \quad 1]$, there is isotropic growth, while for any other case there is anisotropic (transversely isotropic) growth.

Defining the remodeling deformation gradient of constituent i, $\mathbf{F}_r^i(\mathrm{s})$, depends on whether remodeling is occurring in the matrix or in the fibers of the mixture. For the former, there is no direct equation for the gradient, meaning that assumptions need to be made in order to have enough independent equations for the number of unknowns, which must be solved numerically (the reader is redirected to reference [35] for more information on this implementation). Remodeling of the fibers is simpler, since the gradient can be defined as:

$$\mathbf{F}_r^i(s) = \lambda_r^i(s)\boldsymbol{M}^i \otimes \boldsymbol{M}^i + \frac{1}{\sqrt{\lambda_r^i(s)}}(\mathbf{I} - \boldsymbol{M}^i \otimes \boldsymbol{M}^i) \tag{29}$$

where λ_r^i is the remodeling stretch for the one-directional fiber family i and $\boldsymbol{M}^i$ is the direction of the fiber family i.

As previously mentioned, the model implies a dependency on the time variable s, which means that all time dependent variables need to be iteratively updated using, for example, the forward Euler method. When going through the equations above, it is possible to say that time dependency in the equations is originated in the constituent mass density of constituent i, ρ^i, and the remodeling stretch of constituent i, λ_r^i, which means:

$$\rho^i(s + \Delta s) = \rho^i(s) + \dot{\rho}^i(s)\Delta s \tag{30}$$

$$\lambda_r^i(s + \Delta s) = \lambda_r^i(s) + \dot{\lambda}_r^i(s)\Delta s \tag{31}$$

where Δs is an arbitrarily chosen time increment, $\dot{\rho}^i$ is the mass density rate of constituent i and $\dot{\lambda}_r^i$ is the remodeling stretch rate of constituent i.

The mass density rate is defined as:

$$\dot{\rho}^i(s) = \dot{\rho}_+^i(s) + \dot{\rho}_-^i(s) \tag{32}$$

$$\dot{\rho}_+^i(s) = \frac{\rho^i(s)}{T^i}\left(1 + k_{\sigma+}\frac{\sigma_f^i(s) - \sigma_f^i(0)}{\sigma_f^i(0)}\right) \tag{33}$$

$$\dot{\rho}_-^i(s) = -\frac{\rho^i(s)}{T^i}\left(1 + k_{\sigma-}\frac{\sigma_f^i(s) - \sigma_f^i(0)}{\sigma_f^i(0)}\right) \tag{34}$$

With $\dot{\rho}_+^i$ being the rate of mass density deposition of constituent i, $\dot{\rho}_-^i$ being the rate of mass density degradation of constituent i, T^i being the characteristic decay time of constituent i, σ_f^i being a scalarized version of the Cauchy stress, and $k_{\sigma-}$ and $k_{\sigma+}$ being material parameters. Note that expression (32) stands for nonhomeostatic growth, since homeostasis represents a state where $\dot{\rho}^i(s) = 0$. Also note that, when $\sigma_f^i(0) = 0$, the denominator in expressions (33) and (34) is ignored.

On the other hand, the remodeling stretch rate for a fiber constituent i, $\dot{\lambda}_r^i$, is defined as:

$$\dot{\lambda}_r^i(s) = \frac{\frac{\dot{\rho}_+^i(s)}{\rho^i(s)}\left(\sigma_f^i(s) - \sigma_{pre,f}^i(s)\right)\frac{J\phi^i(s)}{4\rho^i(s)}\lambda_r^i(s)}{\frac{\partial^2 W^i(s)}{\left(\partial I_4^i\right)^2}\left(I_4^i(s)\right)^2 + \frac{\partial W^i(s)}{\partial I_4^i}I_4^i(s)} \tag{35}$$

where $\sigma_{pre,f}^i$ is the scalarized version of the homeostatic Cauchy stress $\boldsymbol{\sigma}_{pre}^i(s) = \mathbf{R}(s)\boldsymbol{\sigma}^i(0)\mathbf{R}(s)^T$, such that $\mathbf{F}(s) = \mathbf{R}(s)\mathbf{U}(s)$, where $\mathbf{R}$ is an orthonormal tensor representing the rotational part of $\mathbf{F}$, calculated via polar decomposition, and $\mathbf{U}$ is the right stretch tensor.

2.2.3. Force Balance of a Hollow Cylinder

After having defined the constitutive models for the materials, it is now necessary to perform an equilibrium analysis on the geometry to which the constitutive model is going to be applied. The following analysis is based on references [39, 40].

Thus, let us approximate the geometry of an artery as a hollow thick-walled cylinder of length L, wall thickness H, and inner radius R_i, as Figure 7 indicates. This geometry can be referred to as the reference configuration of the artery, and it is defined in a cylindrical coordinate system.

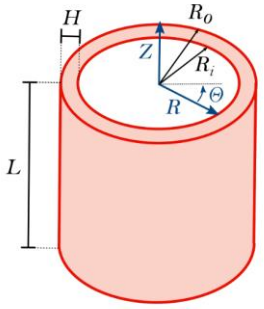

Figure 7 - Schematic of a hollow cylinder [40]

One can start by enunciating the equilibrium equation for solids as:

$$\mathrm{div}(\boldsymbol{\sigma}) + \rho \boldsymbol{b} = \boldsymbol{0} \tag{36}$$

Note that equation (36) defines the force balance on the cylinder under a specific stress state, i.e. in the current configuration (indicated by the lower-case variables, opposed to the upper-case variables of the reference configuration). By neglecting the influence of body forces ($\boldsymbol{b} = \boldsymbol{0}$), it is possible to obtain:

$$\mathrm{div}(\boldsymbol{\sigma}) = \boldsymbol{0} \Leftrightarrow \begin{cases} \dfrac{\sigma_{rr}}{r} + \dfrac{\partial \sigma_{rr}}{\partial r} + \dfrac{1}{r}\dfrac{\partial \sigma_{r\theta}}{\partial \theta} + \dfrac{\partial \sigma_{rz}}{\partial z} - \dfrac{\sigma_{\theta\theta}}{r} = 0 \\ \dfrac{2\sigma_{r\theta}}{r} + \dfrac{\partial \sigma_{r\theta}}{\partial r} + \dfrac{1}{r}\dfrac{\partial \sigma_{\theta\theta}}{\partial \theta} + \dfrac{\partial \sigma_{\theta z}}{\partial z} = 0 \\ \dfrac{\sigma_{rz}}{r} + \dfrac{\partial \sigma_{rz}}{\partial r} + \dfrac{1}{r}\dfrac{\partial \sigma_{\theta z}}{\partial \theta} + \dfrac{\partial \sigma_{zz}}{\partial z} = 0 \end{cases} \tag{37}$$

Then, by assuming axisymmetry of the stresses, one gets the simplified expression:

$$\frac{d\sigma_{rr}}{dr} = \frac{\sigma_{\theta\theta} - \sigma_{rr}}{r} \tag{38}$$

Another way of reaching the balance equation above is by directly performing a stress balance on an infinitesimal section of a cylinder in the radial direction, as is depicted in Figure 8.

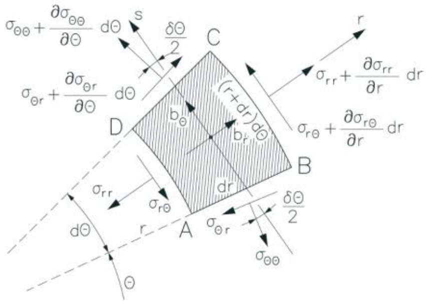

Figure 8 - Infinitesimal part of thick-walled cylinder and corresponding differential stresses [40]

By integrating equation (38) in reference to r, one obtains:

$$P = \int_{r_i}^{r_o} \frac{\sigma_{\theta\theta} - \sigma_{rr}}{r} dr \quad (39)$$

where $P = p_i - p_o$, which is the difference between the pressured being applied on the interior and exterior of the cylinder wall. Note that, in truth, P can also be interpreted as the difference between the symmetric values of the radial stresses being applied on both sides of the wall; in other words, $p_i = -\sigma_{rr}(r_i)$ and $p_o = -\sigma_{rr}(r_o)$ since the radial stresses at the boundary conditions are a reaction to the pressure being applied.

With these equations, it is now possible to start establishing a relation between the undeformed and the deformed/stressed geometry. For example, let us assume biaxial loading (pressure and axial stretch) of the cylinder, then the deformation gradient $\mathbf{F}$ is defined similarly as in equation (5):

$$\mathbf{F} = \begin{bmatrix} \lambda_{rr} & 0 & 0 \\ 0 & \lambda_{\theta\theta} & 0 \\ 0 & 0 & \lambda_{zz} \end{bmatrix} \quad (40)$$

where, by the definition of deformation stretch, and, assuming incompressibility ($\lambda_{zz}\lambda_{\theta\theta}\lambda_{rr} = 1$), we have:

$$\lambda_{zz} = \frac{l}{L} \quad (41)$$

$$\lambda_{\theta\theta} = \frac{r}{R} \quad (42)$$

$$\lambda_{rr} = \frac{1}{\lambda_{\theta\theta}\lambda_{zz}} \quad (43)$$

In expression (42), there is a relation between the undeformed and deformed radii which cannot be discarded, which is revealed via the volume balance between the two configurations:

$$V = v \Leftrightarrow r = \sqrt{\lambda_{\theta\theta,i}^2 R_i^2 + \frac{R^2 - R_i^2}{\lambda_{zz}}} \Leftrightarrow R = \sqrt{R_i^2 + \lambda_{zz}\left(r^2 - \lambda_{\theta\theta,i}^2 R_i^2\right)} \quad (44)$$

With equation (44), an unknown is reached, which is the circumferential stretch at the inner wall of the cylinder $\lambda_{\theta\theta,i}$. This parameter is obtained via expression (39), through an optimization procedure of the following objective function:

$$\min_{\lambda_{\theta\theta,i}} P_{model}(\lambda_{\theta\theta,i}) - P = \int_{r_i}^{r_o} \frac{\sigma_{\theta\theta}(\lambda_{\theta\theta,i}) - \sigma_{rr}(\lambda_{\theta\theta,i})}{r} dr - P \qquad (45)$$

Considering this, the deformation gradient has now been fully defined and is ready to be used in the computation of the stress tensors. It is worth noting that, since an incompressible formulation was assumed, the additional stiffness component that was mentioned in the previous subsection needs to be added to the stress tensor. In order to calculate the Lagrange multiplier, one needs to start by revisiting equation (39) and rewriting the integral from a definite to an indefinite form:

$$(\sigma_{rr}{}^{pre}(r) - p(r)) - (\sigma_{rr}{}^{pre}(r_i) - p(r_i)) = \int_{r_i}^{r} \frac{(\sigma_{\theta\theta}{}^{pre} - p) - (\sigma_{rr}{}^{pre} - p)}{\rho} d\rho \qquad (46)$$

where ρ is simply an auxiliary integration variable and "pre" indicates the stresses before they are combined with the Lagrange multiplier. Since it is known that $\sigma_{rr}{}^{pre}(r_i) - p(r_i) = \sigma_{rr}(r_i) = -p_i$, then, the Lagrange multiplier gets defined in function of the deformed radius as:

$$p(r) = \int_{r_i}^{r} \frac{\sigma_{\theta\theta}{}^{pre} - \sigma_{rr}{}^{pre}}{\rho} d\rho - p_i \qquad (47)$$

2.2.4. Finite Element Method

2.2.4.1. Algorithm

Before finalizing the subsection on constitutive modeling, and since the project at hand also involved the usage of the Abaqus software, it is important to refer the basic theory behind one of the most widely used numerical methods for solving continuum mechanics problems: the Finite Element Method (FEM). The following clarifications on the employment of this method for hyperelasticity were extracted from reference [41].

In the FEM, compared to a standard analysis of a continuum mechanics problem, the stress equilibrium equation gets replaced by the equivalent principle of virtual work. Therefore, the equation for virtual work defined in terms of the Kirchoff stress (in Einstein notation) states that:

$$\int_{R_0} \tau_{ij} \delta L_{ij} dV_0 - \int_{R_0} \rho_0 b_i \delta v_i dV_0 - \int_{\partial R} t_i^* \delta v_i dA_0 = 0 \qquad (48)$$

where $\tau_{ij} = J\sigma_{ij}$ is the Kirchoff stress, δv_i is the virtual velocity fields, δL_{ij} is the virtual velocity gradients, ρ_0 is the mass density in the reference configuration, $\int_{R_0}$ is a volume integral in the reference configuration and $\int_{\partial R}$ is an area integral in the deformed configuration and t_i^* is the traction (force per unit of deformed area).

The procedure for hyperelasticity follows the typical FEM workflow. First, the displacement and virtual velocity fields for an arbitrary point is defined via interpolation between nodes:

$$u_i(\boldsymbol{x}) = \sum_{a=1}^{n} N^a(\boldsymbol{x}) u_i^a \tag{49}$$

$$\delta v_i(\boldsymbol{x}) = \sum_{a=1}^{n} N^a(\boldsymbol{x}) \delta v_i^a \tag{50}$$

where n is the number of nodes that the model has, $\boldsymbol{x}$ is an arbitrary point in the reference configuration, and $N^a(\boldsymbol{x})$ are the nodal shape functions. The deformation for a given displacement is defined as:

$$F_{ij} = \delta_{ij} + \frac{\partial u_i}{\partial x_j} = \delta_{ij} + \sum_{a=1}^{n} \frac{\partial N^a}{\partial x_j} u_i^a \tag{51}$$

Let us now compute the derivatives of the shape functions in terms of the local element coordinates ξ_i. To do this, we have that:

$$x_i = \sum_{a=1}^{N_e} N^a(\xi_j) x_i^a \tag{52}$$

$$\eta_{ij} = \frac{\partial x_i}{\partial \xi_j} = \sum_{a=1}^{N_e} \frac{\partial N^a}{\partial \xi_j} x_i^a \Leftrightarrow \frac{\partial N^a}{\partial x_j} = \frac{\partial N^a}{\partial \xi_j} \eta_{kj}^{-1} \tag{53}$$

where η_{ij} is the Jacobian matrix, and N_e is the number of elements. Then one computes the virtual velocity gradient as:

$$\delta L_{ij} = \frac{\partial \delta v_i}{\partial y_j} = \frac{\partial \delta v_i}{\partial x_k} \frac{\partial x_k}{\partial y_j} = \frac{\partial \delta v_i}{\partial x_k} F_{kj}^{-1} = \sum_{a=1}^{n} \frac{\partial N^a}{\partial x_k} F_{kj}^{-1} \delta v_i^a \tag{54}$$

where y_j is the coordinates of an arbitrary point in the current configuration.

When replacing all these equations into (48), we obtain a set of n nonlinear equations in n unknowns that must hold for every δv_i^a, which can be solved via an iterative method such as the Newton-Raphson method (equation (55)).

$$\left\{ \int_{R_0} \tau_{ij} \left[F_{pk}\left(u_k^b\right) \right] \frac{\partial N^a}{\partial x_m} F_{mj}^{-1} dV_0 - \int_{V_0} \rho_0 b_i N^a dV_0 - \int_{\partial_2 R_0} t_i^* N^a \eta dA_0 \right\} \delta v_i^a = 0 \tag{55}$$

The Newton-Raphson method starts with an initial estimate for u_i^a, which we can call w_i^a. Then, an attempt is made to bring w_i^a closer to u_i^a by an increment, $w_i^a \longrightarrow w_i^a + dw_i^a$, which needs to satisfy:

$$\int_{V_0} \tau_{ij}\left[F_{pk}\left(w_k^b + dw_k^b\right)\right] \frac{\partial N^a}{\partial x_m}(F+dF)_{mj}^{-1} dV_0 - \int_{V_0} \rho_0 b_i N^a dV_0 - \int_{\partial_2 V_0} t_i^* N^a (\eta + d\eta) dA_0 = 0 \qquad (56)$$

Since it's not possible to solve the above equation in reference to dw_k^b, it needs to be linearized (which is explained in [41]), obtaining an expression of the type:

$$K_{aibk} dw_i^a + R_i^a - F_i^a = 0 \qquad (57)$$

where K_{aibk} is the stiffness matrix and R_i^a is the force vector, defined as:

$$K_{aibk} = \int_{V_0} C_{ijkl}^e \frac{\partial N^a}{\partial y_j}\frac{\partial N^b}{\partial y_l} dV_0 - \int_{V_0} \tau_{ij} \frac{\partial N^a}{\partial y_k}\frac{\partial N^b}{\partial y_j} dV_0 - \int_{\partial_2 V_0} t_i^* N^a \frac{\partial \eta}{\partial w_k^b} dA_0 \qquad (58)$$

$$R_i^a = \int_{V_0} \tau_{ij} \frac{\partial N^a}{\partial y_j} dV_0 \qquad (59)$$

where C_{ijkl}^e is the tangent stiffness tensor, defined as:

$$C_{ijkl}^e = \frac{\partial \tau_{ij}}{\partial F_{km}} F_{lm} = J\sigma_{ij}\delta_{kl} + J\frac{\partial \sigma_{ij}}{\partial F_{km}} F_{lm} \qquad (60)$$

This is where the material model chosen comes into play, as it allows for the definition of σ_{ij} and the computation of its derivative.

The algorithm ends by evaluating the convergence via a metric (for example, the magnitude of dw_k^b). If the solution has yet to converge, then one must return to equation (57) and repeat the process.

2.2.4.2. UMAT and `.inp` file

Another important aspect worth mentioning in relation to Abaqus is the `.inp` file. This text document provides all the necessary information associated with an Abaqus Job, including model geometry, material model, boundary conditions, pretended outputs, meshing, section orientation, etc. Through the material definition, it is possible to implement UMATs (User-Defined Materials) into the Job, which is a Fortran subroutine that allows for the implementation of material models that the standard Abaqus environment does not offer. The main inputs and outputs of the subroutine are listed in Table 1:

Table 1 - Overview of the most important input and output variables in a UMAT subroutine [42]

Input	Description
`PROPS`	Array of size $1 \times n$ containing n material properties that are specified via the Abaqus `.inp` file
`DFGRD0`	Array of size 3×3 that contains the deformation gradient at the beginning of the increment (equal to `DFGRD1` of the previous increment)
`DFGRD1`	Array of size 3×3 that contains the deformation gradient at the end of the increment (equal to `DFGRD0` of the following increment)
Output	**Description**
`STRESS`	Array of size $1 \times d$ containing the stress tensor components at the beginning of the increment and which is updated in the subroutine to contain the stress at the end of the increment. For solid elements, $d = 6$.
`DDSDDE`	Array of size $d \times d$ containing the tangent stiffness tensor components which is calculated for the given constitutive model (as seen in (60)). For solid elements, $d = 6$.

Another aspect of the UMAT that should be highlighted is the use of coordinate systems. All deformation gradients map position vectors from one configuration to another. Since these vectors are defined in reference to coordinate systems (the same one for both or two different systems), then the deformation gradient will depend on these coordinate systems as well. In Abaqus, for solid continuum elements, each increment passes a deformation gradient that maps vectors from the undeformed configuration in the current local coordinate system to vectors from the deformed configuration in the same coordinate system. This means that, if the `*orientation` keyword is not present in the `.inp` file, then the computed outputs of the UMAT must be done so in reference to the global coordinate system of the mode, requiring an intermediate step to execute the conversion of the deformation gradient explained in [42, 43]. Otherwise, if `*orientation` keyword is present in that file, then no intermediate step is needed since the computed outputs need to be in the current local coordinate system (Figure 9).

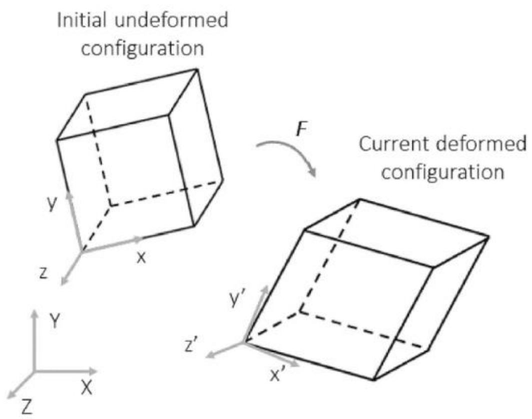

Figure 9 - The three considered coordinate systems. XYZ denotes the global coordinate system, xyz the initial local coordinate system and x'y'z' the current local coordinate system. [42]

2.2.4.3. Hybrid Elements

One final aspect of the FEM that should be mentioned is the type of elements. As previously mentioned, soft tissues possess a highly incompressible behaviour, which can be modeled via a fully incompressible formulation or a near incompressible formulation. In the FEM, the latter involves the already addressed method of adding a volumetric component to the strain energy density function (for example, ${}^{1}/_{D}\,(J-1)^2$), but to employ the former, one must use what are called hybrid elements. Due to the volume not changing under loading ($0.5 \geq v \geq 0.475$), the pressure stress cannot be computed from the displacements of the nodes and, therefore, a pure displacement formulation is inadequate to depict incompressibility, so this type of element needs to be used for fully incompressible behaviour [44].

2.3. Surrogate Modeling

Surrogate Models (SM, or metamodels) serve as approximations to more complex, time consuming and/or computationally intensive models; in other words, they map inputs to their corresponding outputs when the relationship between them is either unknown or too computationally expensive to obtain at higher runtimes [45].

The implementation of surrogates for material models of soft tissues, which is the case in study, has important value in the Uncertainty Quantification (UQ) of said models. When a material model is developed, there are multiple components that possess uncertainty, such as uncertainty derived from approximations and assumptions from the model itself, or from the inputs that go into the model. By knowing the sources, magnitude and propagation of the uncertainty of a model, it is possible to provide guidance in decision-making processes [46]. As such, it is only in an engineer's best interest to be able to perform a UQ framework as efficiently as possible, and, by using a surrogate, it is able to do just that, since it allows to save up computational resources and bypass slow runtimes.

Besides UQ, other applications of surrogates are, for example, in parameter sensitivity analysis, in model calibration, and in decision making [47]. In the latter case, in the medical field, there is interest in creating surrogate models capable of predicting certain outcomes or aiding doctors in reaching a diagnosis, all based on patient data that is readily available at that specific moment. In the particular case of AAA, there have been attempts of creating surrogate models capable of predicting mortality after surgical intervention [13].

According to [48], it is possible to classify surrogate models based on their mathematical structure into three types; data-driven surrogates capture the dominant behaviour of a model by means of input-output samples; in projection-based surrogates the original model is projected into a lower dimensional subspace defined by orthonormal vectors; the third one, multifidelity surrogates, works by reducing the resolution of the outputs, increasing tolerances and overall just simplify the underlying physics behind the model [47].

Delving deeper into data-driven (or machine learned) surrogates, these can be mathematically expressed as:

$$y_{SM} = SM(\boldsymbol{x}, \boldsymbol{\phi}) \approx y = f(\boldsymbol{x}) \tag{61}$$

where SM is the surrogate model function that can approximates its outputs, $\boldsymbol{y}_{SM}$, with the outputs of a constitutive model, $\boldsymbol{y}$, $\boldsymbol{x}$ is the input set and $\boldsymbol{\phi}$ is the hyperaparameter (parameters specific to the model structure) set that creates the best approximation.

The general steps of a constructing a metamodel consist of [49, 50]:

- Choosing the design of experiments to create the training and testing data
- Choosing the model architecture (what type of model to build)
- Mapping the inputs to the outputs of the database (training and validation of the model)
- Assessing the model's quality to replicate the original constitutive model (testing the model)
- Employing the surrogate in the statistical task at hand (i.e., Uncertainty Quantification)

The sections below delve into the process of defining SM by following these general steps.

2.3.1. Design of Experiments

The accuracy of a model is dependent on the method of sampling used in the DoE. Thus, it is important to choose an approach that is simultaneously computationally efficient and allows for a good depiction of the empirical relations of the constitutive model throughout the input variable space.

A commonly used technique is Latin Hypercube Design (LHC), which is a stratified-sampling space-filling type of technique. It is of the former type because it involves the subdivision of the input variables space, and it is also of the latter type because there is a criterion involved in the selection of the sample points [49].

In other words, for an input variable space of size n that is the domain of a given function, and if one wants m samples from said function, then each of the n intervals that define the space are divided into m equal length subintervals, generating m^n hypercubes (subspaces). The criterion for the input selection is such that no two sample points can share a hypercube, nor can the hypercube in which a sample point is inserted share the same subinterval for a given input variable [51, 52]. Adaptations for this method exist such that the length of the subintervals are generated in order to reflect the probability distributions associated with the variables [53]. This method can be visualized better in Figure 10.

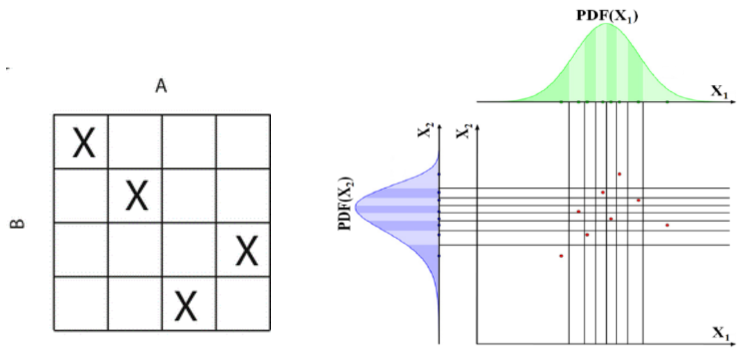

Figure 10 – Examples of random LHC for $n = 2$ and $m = 4$ (left), and gaussian LHC for $n = 2$ and $m = 8$ (right) [52, 54]

Other types of sampling designs include Monte Carlo sampling, Orthogonal design, Quasi-Monte Carlo sampling (and its subvarieties), etc. [49, 51].

2.3.2. Data-driven Model Formats

Many different types of surrogate model formats exist, such as polynomial chaos functions, gaussian process regressions, neural networks, etc. [50]. For this project, neural networks were and gaussian processes were used, and are explained below.

2.3.2.1. <u>Deep Neural Networks</u>

Neural networks are a type of machine learning format based on the way human neurons behave. They are made up of an input layer, a hidden layer and an output layer; each layer performs a linear combination between the input, weights and biases, followed by an activation function that confers nonlinear capabilities to the network. Deep Neural Networks (DNN) are a more robust version of NN that contain multiple hidden layers, allowing for the mapping of more complex problems [55, 56]. The two main uses of NN are for regression (mapping a set of inputs to a real number), or classification problems (mapping a set of inputs to a categorical label).

A NN architecture can be generalized by the expression [57]:

$$\boldsymbol{h}_i = g_i(\mathbf{W}_i \boldsymbol{h}_{i-1} + \boldsymbol{b}_i) \qquad (62)$$

where, for a given layer i of the network, $\boldsymbol{h}_i$ is its output, $\boldsymbol{h}_{i-1}$ is the output of the previous layer that serves as the input of the current layer, $\mathbf{W}_i$ and $\boldsymbol{b}_i$ are the weights and bias of the layer, respectively, and g_i is the nonlinear activation function of the layer. For the case of $i = 1$, $\boldsymbol{h}_{i-1}$ is the input vector of the model, $\boldsymbol{x}$ (Figure 11), and for the last layer, $\boldsymbol{h}_i$ is the output of the surrogate, y_{SM}.

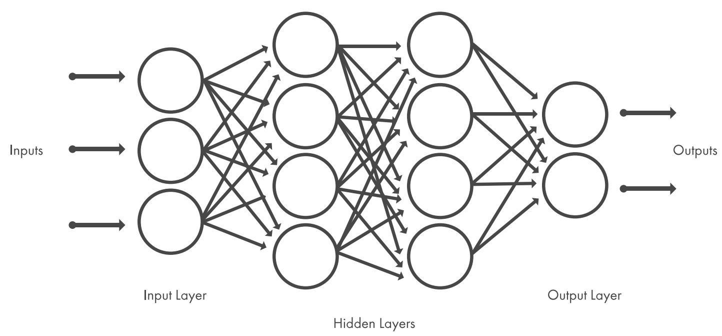

Figure 11 - Typical Neural Network architecture[58]

As such, weights and biases are the attributes of NN that need to be taught in order to perfect the mapping between inputs and the outputs of the dataset (also called collectively as the learnable parameters, $\boldsymbol{\theta}$). This is named the Training of the NN, and, for regression problems (such as ours), it occurs in three phases: forward propagation, Loss computation, and backward propagation (Figure 12) [56, 59].

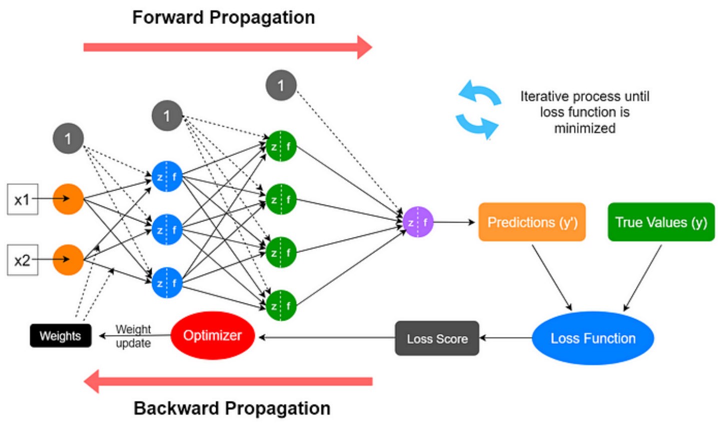

Figure 12 - Framework of NN Training [59]

The first phase involves sending a data batch of size n through the NN in order to compute $\mathbf{Y}_{SM}$ for the given $\mathbf{X}$ of the batch. The second one involves calculating the loss function, which indicates how distant the predictions are from the real values. A typical loss function for regression NN is the Mean Squared Error (MSE):

$$L = MSE(\mathbf{Y}_{SM}, \mathbf{Y}) = \sum_{i=1}^{n} \left(y_{SM_i} - y_i\right)^2 \qquad (63)$$

where n is the batch size and y_i is the output for instance i of the batch. After this, the gradient of the loss function in reference to the learnable parameters is calculated via automatic differentiation. Finally, in the backward propagation phase, the learnable parameters are updated through an algorithm that minimizes the loss function based on the computed gradients. One common algorithm to update the learnables is the ADAM (Adaptive moment estimation) [59, 60]. This process repeats itself through multiple iterations. When the entire dataset has been processed, it is said that one epoch has passed in training of the model.

The hyperparameters of NN include every variable involved in its architecture, such as the number of hidden layers, the output size of the layers/number of neurons per layer (which is behind the size of the weights and biases), but also involves variables on how the model is trained (such as learning rate) [57].

One final important aspect of neural networks is weight regularization. This consists in a set of processes that penalize larger weights, allow for a better convergence of the training process and for reduced overfitting (when the model is extremely constricted to predict values using inputs from the training database and is not flexible enough to predict using unseen inputs). Two common ways to apply weight regularization is through input and/or output standardization (transforming these values into a common range), and weight penalties, such as the L2 penalty [61].

2.3.2.2. Physics-Informed

Physics-Informed Neural Networks are a subsect of NN that have been explored and various papers such as [62-64]. The main way to implement physics into the model training is by adding and extra term to the loss function:

$$L = (1-\varepsilon)\sum_{i=1}^{n}\left(y_{SM_i} - y_i\right)^2 + \varepsilon\sum_{i=1}^{n_{PI}}\left(p_{SM_i} - p_i\right)^2 \tag{64}$$

where ε is a parameter that weighs the two terms of the loss, p_i is the value of a physics law to which the surrogate must obey, p_{SM_i} is the actual value for the same physics law obtained via the surrogate, and n_{PI} is the arbitrary number of data instances which were artificially generated to evaluate the physics relation. The interesting aspect about this type of NN is this concept of quickly evaluating artificial inputs during training, generating artificial loss based on a real equation that the model needs to obey.

2.3.2.3. Variational Inference Bayesian Neural Network

Variational Inference Bayesian Neural Networks (VIBNN) are a specific type of Bayesian Neural Networks (BNN), which is used for the purpose of Uncertainty Quantification. Bayesian Neural Networks are a theoretical type of NN based on Bayesian Inference, which states that:

$$p(H|D) = \frac{p(D|H)p(H)}{p(D)} \tag{65}$$

Where $p(H|D)$ is the posterior distribution, $p(H)$ is the prior distribution, $p(D|H)$ is the likelihood distribution, H is the parameter set on which we intend to infer, and D is empirical knowledge that informs our belief about H. In theory, for BNN, H would consist of the network weights and biases and D would be the training dataset, but, due to the high dimensionality of the problem (neural networks have thousands of learnable parameters), and since the posterior distribution is extremely complex to sample from, the alternative solution is the so called (Stochastic) Variational Inference Bayesian Neural Network. In these metamodels, the learnables are actually the means and variances of the weights and biases that make up the NN, which are sampled from in each run. This generates a family of distributions, Q, and, to find the closest member of the family, $q^*(\boldsymbol{\theta})$, that resembles the posterior distribution, $p(\boldsymbol{\theta}|D)$, the most, one needs to maximize the negative ELBO loss (which is based on the Kullback-Leibler divergence between distributions):

$$L(q^*(\boldsymbol{\theta})) = -ELBO(q^*(\boldsymbol{\theta})) = E_{q^*(\boldsymbol{\theta})}[\log(p(\boldsymbol{Y}|\boldsymbol{\theta},\boldsymbol{X})p(\boldsymbol{\theta}))] - E_{q^*(\boldsymbol{\theta})}[\log(q^*(\boldsymbol{\theta}))] \quad (66)$$

UQ is possible using this type of model since, if the model processes the same input multiple times, it is possible to obtain a probability distribution of the output (each time the input is processed, the weights get resampled, thus generating a different output each turn) [65-67].

2.3.2.4. Gaussian Process Regression

Gaussian Process Regressions (GPR, also called kriging) are a format of metamodels of the type:

$$y_{SM} = h(\boldsymbol{x})^T\beta + f(\boldsymbol{x}) \quad (67)$$

where $f(\boldsymbol{x})$ is a Gaussian Process (meaning that the joint distribution of $f(\boldsymbol{x}_1)$, $f(\boldsymbol{x}_2)$, …, $f(\boldsymbol{x}_n)$, where n is the number of input sets of the training database, is normal with mean zero)), $h(\boldsymbol{x})$ is a vector of basis functions and β the base function coefficients. The key aspect about GP is its stochastic nature, meaning it that given an input set, the GP provides the output mean, as well as its variance (and thus a probability density function for the output) [47, 68, 69]. The hyperparameters of this model include the kernel function (also called the covariance function), the basis functions, or the kernel scale.

2.3.3. Performance Metrics

Following the training, as previously said, the surrogate needs to undergo an evaluation of its mapping performance. A common evaluation metric when evaluation a surrogate model (either after or during training) is the Root Mean Square Error:

$$RMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(y_{SM_i} - y_i\right)^2} \quad (68)$$

The most important metric to evaluate the surrogate's capabilities is the R-Squared (or Coefficient of determination) metric, defined between zero and one. Although there is no standardized consensus, references point towards $R^2 \geq 0.7$ or 0.8 and higher as good values for the surrogate [70]:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_{SM_i} - y_i\right)^2}{\sum_{i=1}^{n}(\bar{y} - y_i)^2} \tag{69}$$

where $\bar{y}$ is the average of the true value of the true response and n is the number of instances in the training dataset.

2.4. Uncertainty Quantification

Computational modeling has become extremely relevant in engineering fields in order to represent real-life physics phenomena, as it has been mentioned previously. But these models, after being formulated, present challenges. When formulating these models, assumptions might have been made, or parameters were incorporated without much information on their accuracy. In other words, there is a necessity to increase the reliability of computational models, since they don't always faithfully describe the behaviour exhibited by experimental setups. Therefore, Uncertainty Quantification seeks to assess the credibility of *in silico* models by means of identifying the sources of uncertainty, developing evaluation metrics, and quantifying their impact in the model's outcomes [50].

When it comes to types of uncertainty, one can identify the following ones [46]:

- Aleatoric uncertainty: originates due to the random nature of the event/variable. This means that, by sampling the given variable a sufficient number of times, the uncertainty associated with it is decreased. It is characterized by a probability density function, or a cumulative distribution function (CDF).
- Epistemic uncertainty: derives from lack of knowledge on the variable/event. By increasing the information behind the variable, it is possible to decrease the uncertainty, in theory. It is characterized in the form of an interval, where no value within it is more likely to occur than the other.

This classification the relation between these types of uncertainty can be visualized in Figure 13. The interesting aspect to note is that, when parameters possess both types of uncertainty, then that implies that there is uncertainty behind the probability distribution that should define it (imprecise probability) [50].

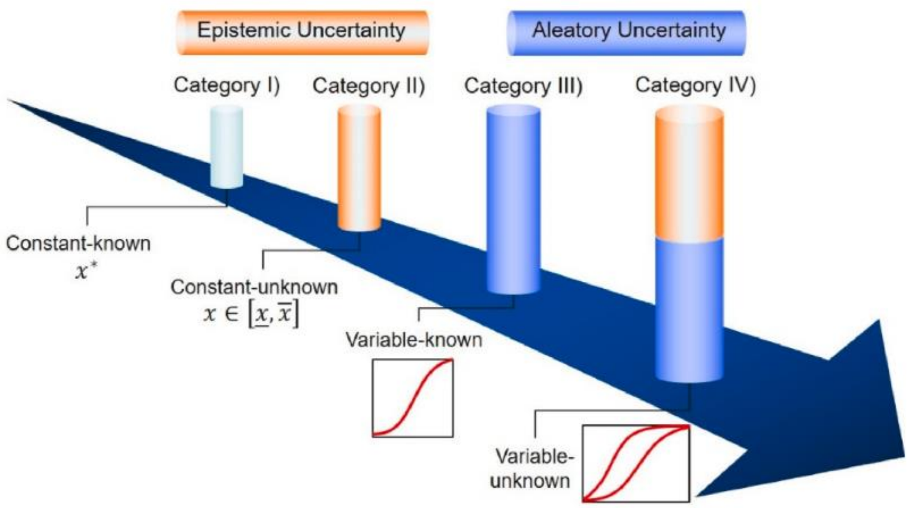

Figure 13 - Categorization of parameters according to different types of uncertainty [50]

When it comes to uncertainty sources, they may arise from all different kinds of components that are incorporated into the computational model, such as inputs (geometry, loading, material parameters, etc.), numerical approximations (due to rounding errors or the computer's rounding capabilities), or from the model form (assumptions adopted to formulate the model, such as numerical iterations to solve complex equations) [46].

But, performing UQ frameworks in heavy and expensive to evaluate constitutive models may be practically impossible, which is why, as it was previously mentioned, metamodels have gained popularity to replicate the behaviour of the original models, and thus, replace them in the analysis of their uncertainty [49].

3. Methodology

This section focuses on the decision-making process of the project, as well as the design of experiments (DoE) behind the obtained results in the following section. In this way, it is divided into a first subsection that describes the constitutive models (CM) used to represent the mechanical behaviour of an artery under loading, and a second subsection that details how data was extracted from said models and employed for the training of the surrogates. The third subsection lists the models that were trained in the scope of the surrogate training and the quantification of their uncertainty.

It should also be highlighted that the developed code was done so with the support of the MATLAB Documentation, which is available in [71-76], and the Abaqus Documentation, available in [44, 77].

3.1. Constitutive Models

3.1.1. MATLAB Models

For the project at hand, two types of MATLAB models were created to represent arterial behaviour. For each one, code was created to extract the stresses and deformed radii given a set of geometric, material, structural and loading inputs.

The models themselves are based on the formulation presented in section 2.2.3. All models are called WSE (Wall Stress Estimator) and are numbered from 1 to 3 in order of complexity, and, additionally, the following assumptions are common to all of them:

- Cylindrical coordinate system ($r - \theta - z$).
- Axisymmetry
- Incompressibility
- Thick-walled cylinder
- The pressure on the outer wall is null ($p_o = 0$)

3.1.1.1. WSE1

The first of the models pretends to model the stress behaviour of an artery as a thick-walled cylinder under biaxial loading using a Neo-Hookean material model. Tables 2 and 3 list the inputs and outputs of the model, Figure 14 portrays the loading the cylinder is under, and equation (70) represents the deformation gradient representative of said loading.

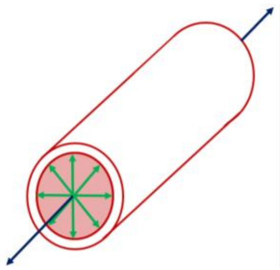

Figure 14 - Schematic for WSE1

$$\mathbf{F} = \begin{bmatrix} \frac{1}{\lambda_{\theta\theta}\lambda_{zz}} & 0 & 0 \\ 0 & \lambda_{\theta\theta} & 0 \\ 0 & 0 & \lambda_{zz} \end{bmatrix} \qquad (70)$$

Table 2 - Listing of inputs for WSE1

Input	Name	Type
H [mm]	Undeformed wall thickness	Geometric
R_i [mm]	Undeformed inner radius	Geometric
λ_{zz} [-]	Axial stretch	Loading
C_{10} [MPa]	Neo-Hookean parameter	Material
p_i [MPa]	Pressure exerted on inner wall	Loading

Table 3 - Listing of outputs for WSE2

Output	Name
$\boldsymbol{\sigma}$ [MPa]	Cauchy stress tensor
r_i [mm]	Deformed inner radius
r_o [mm]	Deformed outer radius

3.1.1.2. WSE2

The second model predicts the stress behaviour of an artery as a thick-walled, two-layered cylinder (tunica media and adventitia) under biaxial loading using a GOH material model, with two fiber families. Tables 4 and 5 list the inputs and outputs of the model, Figure 15 portrays the loading the cylinder is under, and equation (72) represents the deformation gradient representative of said loading.

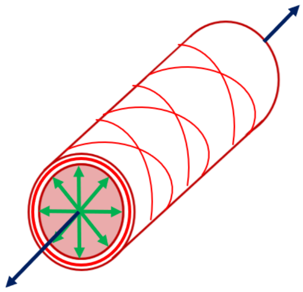

Figure 15 - Schematic for WSE2

$$\mathbf{F} = \begin{bmatrix} \frac{1}{\lambda_{\theta\theta}\lambda_{zz}} & 0 & 0 \\ 0 & \lambda_{\theta\theta} & 0 \\ 0 & 0 & \lambda_{zz} \end{bmatrix} \qquad (71)$$

Table 4 - Listing of inputs for WSE2

Input	Name	Type	Layer
H_M [mm]	Undeformed thickness	Geometric	Tunica Media
H_A [mm]	Undeformed thickness	Geometric	Tunica Adventitia
R_i [mm]	Undeformed inner radius	Geometric	-
λ_{zz} [-]	Axial stretch	Loading	-
C_{10_M} [MPa]	Neo-Hookean parameter	Material	Tunica Media
κ_M [-]	GOH parameter	Material	Tunica Media
k_{1_M} [MPa]	GOH parameter	Material	Tunica Media
k_{2_M} [-]	GOH parameter	Material	Tunica Media
α_M [rad]	Fiber family orientation angle	Structural	Tunica Media
C_{10_A} [MPa]	Neo-Hookean parameter	Material	Tunica Adventitia
κ_A [-]	GOH parameter	Material	Tunica Adventitia
k_{1_A} [MPa]	GOH parameter	Material	Tunica Adventitia

k_{2_A} [-]	GOH parameter	Material	Tunica Adventitia
α_A [rad]	Fiber family orientation angle	Structural	Tunica Adventitia
p_i [MPa]	Pressure exerted on inner wall	Loading	-

Table 5 – Listing of outputs for WSE2

Output	Name	Layer
$\boldsymbol{\sigma}_M$ [MPa]	Cauchy stress tensor	Tunica Media
$\boldsymbol{\sigma}_A$ [MPa]	Cauchy stress tensor	Tunica Adventitia
r_i [mm]	Deformed inner radius	-
r_m [mm]	Deformed middle radius	-
r_o [mm]	Deformed outer radius	-

3.1.2. *Abaqus Models*

Besides the MATLAB code, Abaqus python scripts were also developed to simulate arterial behaviour. Three types of arterial models for Abaqus were created: the first two are identical in terms of formulation to the WSE1 and WSE2 previously explained, with the added material parameter $D = 0$ since we assume incompressibility. In more detailed terms, these Abaqus models involve a 3D axisymmetric model of a tube quarter, where pressure is applied in a surface defined by the inner wall, axial extension is applied in one of the axial sides, and the remaining sides of the artery have their displacements constrained in their perpendicular. The simulation is computed in one Abaqus/Standard step, with Nlgeom turned on. For WSE2, a partition is made to differentiate between the tunica media and adventitia layers, to allow the assignment of sections with different material properties. The elements used were C3D8H and the element size of 0.1 mm.

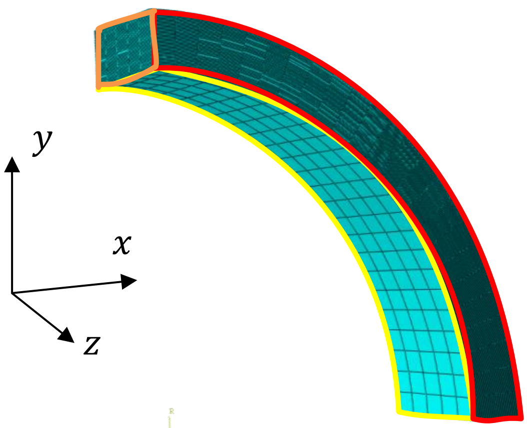

Figure 16 - Quarter Artery of the Abaqus model. Region in red is the Set where axial displacement is applied; region in yellow is the Surface where pressure is applied; region in orange is the Set where displacement in XX is constrained. Not seen: the Sets where displacements in YY and ZZ are constrained.

The third model (called WSE3) is a growth and remodeling UMAT that holds the same assumptions that were mentioned at the beginning of subsection 3.1.1, with the exception of incompressibility, since near-incompressibility is assumed (Figure 17). It is divided into three steps: a first one that serves to apply a homeostatic stress to the inner wall of the cylinder, a second one where a prestressing algorithm is executed to find the homeostatic configuration (configuration at which the cylinder is not deformed but possesses residual stresses due to the first step), and a third step where non-homeostatic growth and remodeling of four fiber families is considered for 100 time increments. The type of element used in the simulation is the C3D8. The reasoning behind this UMAT is explained in more detail in [35, 78].

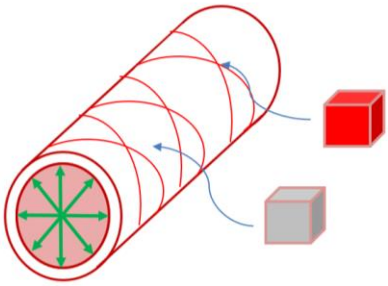

Figure 17 - Schematic of WSE3

3.1.1. Abaqus Mesh Size Convergence

For all three Abaqus models, a mesh size convergence test was performed. In this test, the size of the elements in the radial direction was varied between 0.025 and 0.4 mm, from the initial reference of 0.1 mm. The percentage of variation of the radial and tangential stresses at the extremity of the model was registered (Figure 18). Results are presented in subsection 4.1.1.

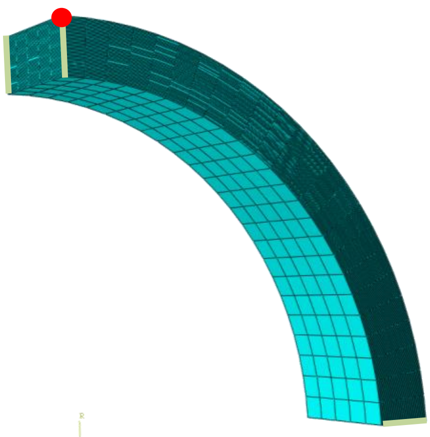

Figure 18 – In red is the node where the stresses were extracted from for the mesh convergence test, and in green are the Sets where bias seeding was applied to obtain the pretended element size in the radial direction

3.1.2. *MATLAB & Abaqus Comparison*

In order to cross validate the code formulated in MATLAB and Abaqus, a comparison between the stress outputs from both types of models was made. Tables 6 and 7 list the input sets for WSE1 and WSE2 that were used for the comparison. Results are presented in subsection 4.1.2.

Table 6 - Input values for WSE1

Input	Value
H [mm]	2.5
R_i [mm]	15
λ_z [-]	1.4
C_{10} [MPa]	0.5
p_i [MPa]	0.02

Table 7 - Input values for WSE2

Input	Value
H_M [mm]	1.25
H_A [mm]	1.25
R_i [mm]	15
λ_z [-]	1.4
C_{10_M} [MPa]	0.05
k_{1_M} [MPa]	0.34
k_{2_M} [-]	7.2
κ_M [-]	0.8
α_M [rad]	$\pi/8$
C_{10_A} [MPa]	0.06
k_{1_A} [MPa]	0.27
k_{2_A} [-]	13.4
κ_A [-]	0.29
α_A [rad]	$\pi/4$
p_i [MPa]	0.02

3.2. Design of Experiments

After having the models defined, one needs to establish a discrete correspondence between different input sets and their outputs, which will allow for an empirical description of the models. This procedure is called sampling, and it serves to generate the data necessary to train the data-driven surrogates.

In this project, two ways of collecting the necessary data were defined: one for WSE1 & WSE2 and another for WSE3. For the former, the MATLAB version of the models was used for sampling due to lower processing time compared to Abaqus, and two types of outputs were considered to be sampled: the deformed inner radius, r_i, and the stress tensor, $\boldsymbol{\sigma}$. In the case of WSE3, the Abaqus model was used, and the output considered to be sampled was the tangential stress, σ_θ. These approaches are explained in more detail below.

3.2.1. Sampling the Deformed inner radius for WSE1 & WSE2

Firstly, one needs to define the input variable space (IVS) for each constitutive model. In other words, to define the upper and lower boundaries of each possible input, the following expressions were used:

$$lb_i = refval_i \times (1 - val) \tag{72}$$

$$ub_i = refval_i \times (1 + val) \tag{73}$$

where, for each input i, lb_i is the lower boundary, ub_i is the upper boundary, $refval_i$ is a reference value based on literature of the different inputs ([79-83]) and val is an arbitrary value. For all variables, it was postulated $val = 0.2$, which means the upper and lower boundaries are a mere percentual increase and decrease of 20% from the reference value, respectively.

The next step requires the definition of an expanded input variable space (EIVS), and an unseen input variable space (UIVS), which will be used for testing the surrogate. Testing the surrogate on inputs it was not trained with allows for a better understanding of its capability for generalization and of how well it depicts the constitutive model relations between inputs. For the case of the EIVS boundaries, we have:

$$elb_i = lb_i - \frac{ub_i - lb_i}{2} \times perc = lb_i - refval_i \times val \times perc \tag{74}$$

$$eub_i = ub_i + \frac{ub_i - lb_i}{2} \times perc = ub_i + refval_{\theta_2} \times val \times perc \tag{75}$$

where $perc$ is a parameter that serves to expand the boundaries, elb_i is the expanded lower boundary and eub_i is the expanded upper boundary. This implies that the UIVS boundaries are defined as:

$$ulb1_i = elb_i \tag{76}$$

$$uub1_i = lb_i \tag{77}$$

$$ulb2_i = ub_i \tag{78}$$

$$uub2_i = eub_i \tag{79}$$

where $ulb1_i$ and $uub1_i$ are the unseen lower and upper boundaries on one side, respectively, and $ulb2_i$ and $uub2_i$ are the unseen lower and upper boundaries on the other side, respectively.

For this project, three values of $perc$ were chosen (5, 2, and 1), meaning that three datasets were created for UIVS and EIVS.

To better understand the expressions presented, a simple example is presented: let f be the function that mathematically represents a certain constitutive model such that:

$$y = f(\boldsymbol{\theta}) \tag{80}$$

where $\boldsymbol{\theta} = [\theta_1 \quad \theta_2]$ is the input vector made up of two scalars. This means that the input variable space $IVS \subset \mathbb{R}^2$ and that the expanded input variable space $EIVS \subset \mathbb{R}^2$. Let us assume $refval_{\theta_1} = 1$ and $refval_{\theta_2} = 0.4$, and, for $perc = 1$, IVS and $EIVS$ can be visually represented as Figure 19 suggests.

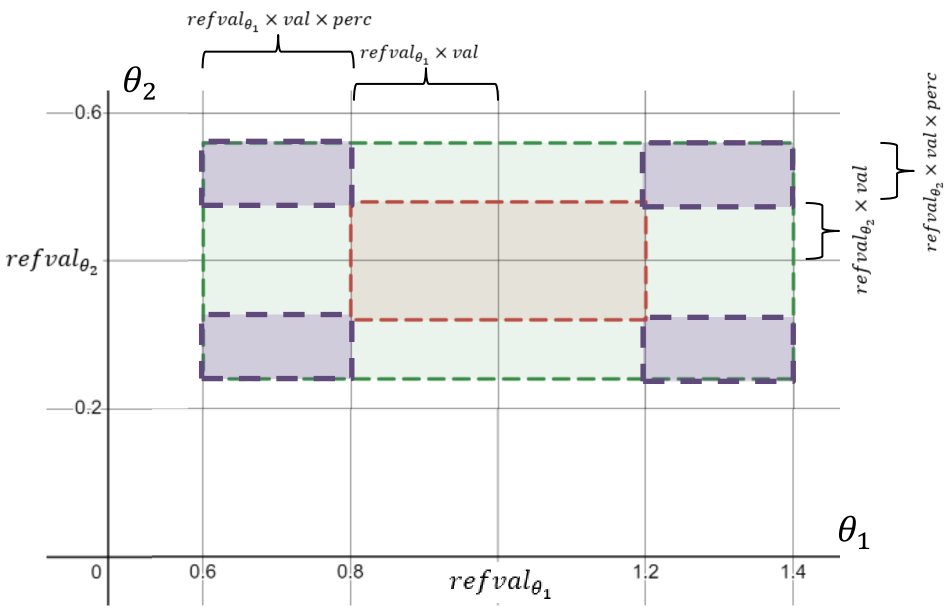

Figure 19 - Input variable spaces, where green is EIVS, purple is UIVS and red is IVS ($IVS \subset EIVS \subset \mathbb{R}^2, UIVS \subset EIVS \subset \mathbb{R}^2$) (image created with Desmos)

The above train of thought can be generalized for $\mathbb{R}^n$, where n is the size of the input variable vector. As such, the reference values that were used for WSE1 and WSE2 are presented in Tables 8 and 9.

Table 8 - Reference values for the inputs of WSE1

Input	Reference values, $\boldsymbol{refval}$
H [mm]	1
R_i [mm]	13
λ_z [-]	1.1
C_{10} [MPa]	0.5
p_i [MPa]	0.03

Table 9 - Reference values for the inputs of WSE2

Input	Reference values, $\boldsymbol{refval}$
H_M [mm]	1
H_A [mm]	1
R_i [mm]	13
λ_z [-]	1.1
C_{10_M} [MPa]	0.05
k_{1_M} [MPa]	0.2
k_{2_M} [-]	0.3
κ_M [-]	6
α_M [rad]	$\pi/8$
C_{10_A} [MPa]	0.05
k_{1_A} [MPa]	0.2
k_{2_A} [-]	0.3
κ_A [-]	14
α_A [rad]	$\pi/4$
p_i [MPa]	0.03

With the input spaces defined, one needs to adopt an algorithm that chooses points from the defined variable spaces, which, for this project, was Random Latin Hypercube. Applying this method to sample from IVS and EIVS is straightforward; to sample from UIVS, however, the adopted approach was to sample from the region defined by $ulb1_i$ and $uub1_i$ (in the previous example, it would be the lower left purple square), and by $ulb2_i$ and $uub2_i$ (upper right purple square), and then randomly permute the sampled values for the inputs in order to achieve sampling of the remaining UIVS (upper left and lower right purple squares). For all datasets, the number of sampled instances was 1000.

3.2.2. Sampling the Stress Tensor for WSE1 & WSE2

When it comes to the stress tensor, another input arises, which is the r where we want to extract the stresses from. Since this parameter will vary depending on the geometry, a normalized version of the model was adopted:

$$r_{norm} = \frac{r - r_l}{r_h - r_l} \in [0 \quad 1] \tag{81}$$

where r_l is the lower deformed radius of the layer, and r_h is the higher deformed radius of the layer. Keep in mind that, since WSE2 incorporates two material layers, two stress tensors need to be extracted, which means that, for the tunica media, $r_l = r_i$ and $r_h = r_m$, while for the tunica adventitia, $r_l = r_m$ and $r_h = r_o$. The reference value for r_{norm} was adopted as 0.5.

3.2.3. Sample Rejection in WSE1 & WSE2

During sampling, especially when sampling from the EIVS and the UIVS, certain input sets produce unrealistic and/or problematic results that contaminate the dataset. Thus, it is important to remove these instances based on a set of criteria. For this reason, if, for a given input set…

- Outputs took longer than 30 seconds to compute
- No numerical solution of equation (45) for the variable $\lambda_{\theta\theta,i}$ within the range $\left[{}^{1}\!/_{3} \quad 3\right]$ was found
- Boundary conditions were not met (i.e., $\sigma_{rr}(r_o) = 0$, and $\sigma_{rr}(r_i) = -p_i$)
- The computed deformed inner radius, r_i, is bigger than 60 mm
- No solution for the Lagrange Multiplier was found

…then the input set was rejected. The amount of rejected inputs is detailed in subsection 4.2.1. It should be noted that no samples were rejected when sampling for the IVS (i.e., all instances in the IVS fulfilled the above criteria).

3.2.4. Output Statistics in WSE1 & WSE2

After obtaining the datasets, a general view of the distribution of the outputs is provided in subsection 4.2.2.

3.2.5. *Sampling in WSE3*

In terms of sampling in WSE3, the approach was more unorthodox. Instead of using an LHC framework using a MATLAB model, the G&R simulation was executed in Abaqus with specific material and geometric parameters listed in Table 10, based on [35]:

Table 10 – Set of parameters used in the Abaqus simulation of WSE3

Input	Values
C_{10} [MPa]	0.0305/0.8
k_1 [MPa]	0.0289/0.05
R_i [mm]	5
k_2 [-]	1.23
α [rad]	$\pi/8$
H [mm]	1.3
$\rho^{elastin}(0)$ [-]	0.8
$\rho^{collagen,j}(0)$ [-]	0.05
$g_{zz}^{elastin}$ [-]	1.2
$g^{collagen,j}$ [-]	1.1
$T^{collagen,j}$ [days]	101
$k_{\sigma+}^{collagen,j}$ [-] or [MPa^{-1}]	0.1
$k_{\sigma-}^{collagen,j}$ [-] or [MPa^{-1}]	0.0
p_i [MPa]	0.03

The resulting `.odb` file contains, in its third time step, the G&R phase over the course of 100 time increments (equivalent to 1000 days of G&R). The sampling procedure consisted in extracting the values for the tangential stress for each increment for each radial position; in other words, the resulting data set had as inputs the radius and the day, and as the output the tangential stress. Finally, the data was partitioned into two: one that contained all data regarding the first 30 increments (first 300 days) for training, and the remaining data for testing.

3.3. Surrogate Models

To provide the reader with a better overview of the created figures, Tables 11 to 13 list all the surrogates that were built and the studies that were executed with them, along with the associated figures in section 4.

Table 11 - Surrogates for WSE1

Name	Type	Output	Types of Figures & Tables	Reference
WSE1SM1	GPR	r_i	Predicted VS Real Values Plot	Figure 37
			R^2 VS $perc$ Plot	Figure 38
			PDF for UQ	Figure 39
			Partial Dependence Plots	Figures 40-42
			Hyperparameters	Table 17
WSE1SM2	NN	r_i	Predicted VS Real Values Plot	Figure 43
			R^2 VS $perc$ Plot	Figure 44
			Training Set Size VS R^2 Plot	Figure 45
			Hyperparameters	Table 18

Table 12 - Surrogates for WSE2

Name	Type	Output	Types of Figures & Tables	Reference
WSE2SM1	GPR	r_i	Predicted VS Real Values Plot	Figure 46
			R^2 VS $perc$ Plot	Figure 47
			PDF for UQ	Figure 48
			Partial Dependence Plots	Figures 49-51
			Hyperparameters	Table 19
WSE2SM2	NN	r_i	Predicted VS Real Values Plot	Figure 52
			R^2 VS $perc$ Plot	Figure 53
			Training Set Size VS R^2 Plot	Figure 54
			Hyperparameter Sensitivity	Figures 55-57
			Hyperparameters	Table 20
WSE2SM3	NN	r_i, r_m, r_o, σ_{rr_M}, $\sigma_{\theta\theta_M}$, σ_{rr_A}, σ_{rr_M}	Predicted VS Real Values Plot	Figures 58-64
			R^2 VS $perc$ Plot	Figures 65 and 66

WSE2SM4	PINN	r_i, r_m, r_o, σ_{rr_M}, $\sigma_{\theta\theta_M}$, σ_{rr_A}, σ_{rr_M}	Predicted VS Real Values Plot	Figures 67-73
			R^2 VS $perc$ Plot	Figures 74 and 75
			ε Sensitivity	Figure 76
WSE2SM5	VIBNN	r_i	Predicted VS Real Values Plot	Figure 77
			PDF for UQ	Figure 78
			Hyperparameters	Table 20

Table 13 - Surrogate for WSE3

Name	Type	Output	Study	Figures & Tables
WSE3SM1	GPR	$\sigma_{\theta\theta_M}$	Predicted VS Real Values Plot	Figure 79
			Hyperparameters	Table 24

Before proceeding with the results, a few clarifications should be made on the models. First, it should be noted that the GPRs and WSE1SM1 were trained using the built-in MATLAB Regression Learner app. Additionally, from WSE2SM4 through WSE2SM5, the same hyperparameters were used.

In terms of standardization of the dataset, symmetric standardization ($x_i^{stand} = 2 \times \frac{x_i - lb_i}{ub_i - lb_i} - 1$) was used for the inputs, while the outputs went through z-score standardization ($y_i^{stand} = \frac{y_i - \mu_i}{\sigma_i}$).

When it comes to the PINN, during training, 300 artificial inputs were generated: 100 for random r_{norm}, 100 for $r_{norm} = 1$ and 100 for $r_{norm} = -1$. The respective outputs were calculated by the NN and then their outputs were incorporated into the equations in the following way:

$$L_1 = \text{MSE}\left(\text{zscore}\left(\left(\frac{\sigma_{\theta\theta_M}^{PINN} - \sigma_{rr_M}^{PINN}}{r_M} - \frac{d\sigma_{rr_M}^{PINN}}{dr_M}\right)\right)\right) \tag{82}$$

$$L_2 = \text{MSE}\left(\text{zscore}\left(\left(\frac{\sigma_{\theta\theta_A}^{PINN} - \sigma_{rr_A}^{PINN}}{r_A} - \frac{d\sigma_{rr_A}^{PINN}}{dr_A}\right)\right)\right) \tag{83}$$

$$L_3 = \text{MSE}\left(\text{zscore}\left(\sigma_{rr_A}^{PINN} - \sigma_{rr_M}^{PINN}\right)\right) \tag{84}$$

$$L_4 = \text{MSE}\left(\text{zscore}\left(\left(-\sigma_{rr_A}^{PINN}\right) - 0\right)\right) \tag{85}$$

$$L_5 = \text{MSE}\left(\text{zscore}\left(\left(-\sigma_{rr_M}^{PINN}\right) - p_i\right)\right) \tag{86}$$

$$L_{Physics} = \frac{1}{5}\sum_{i=1}^{5} L_i \tag{87}$$

where equations (82) and (83) are applied to the 300 outputs, and (84) to (86) are applied to the outputs of PINN where the r_{norm} was 1 or -1 (since the stresses are being evaluated at the extremities of each layer).

When it comes to model validation during training, holdout validation was performed with a holdout percentage of 30%.

4. Results

In this section, results are presented regarding the methodology previously explained. As such, it includes figures and comments associated with the validation of the constitutive models, the design of experiments and, finally, the performance of the metamodels.

4.1. Constitutive Models

4.1.1. Abaqus Mesh Size Convergence

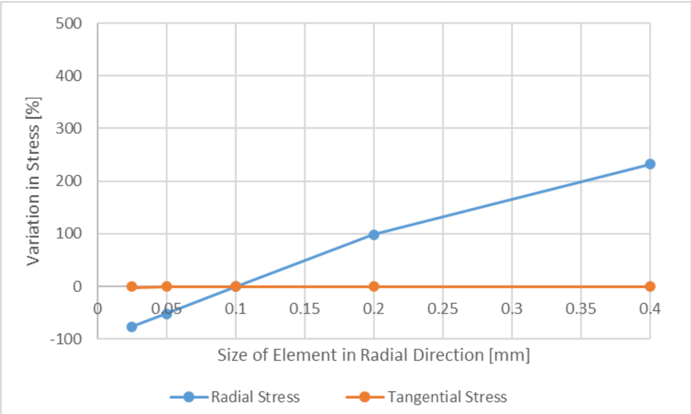

Figure 20 - Mesh Size Convergence Test for WSE1 (blue for Radial Stress, orange for Tangential Stress)

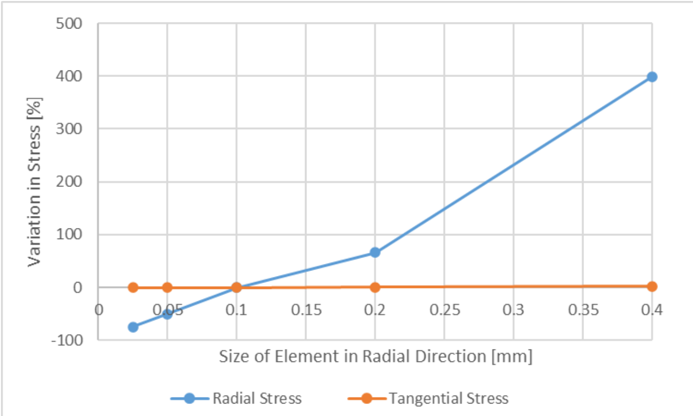

Figure 21 - Mesh Size Convergence Test for WSE2 (blue for Radial Stress, orange for Tangential Stress)

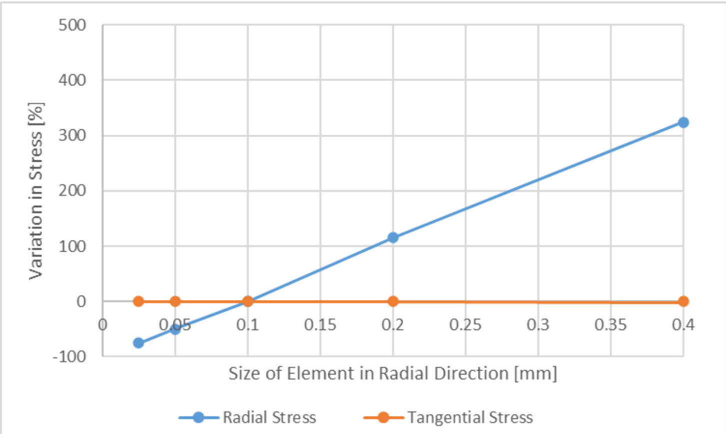

Figure 22 - Mesh Size Convergence Test for WSE3 (blue for Radial Stress, orange for Tangential Stress)

From Figures 20-22, it is possible to observe a linear relationship between the size of the elements in the radial direction and the percentual variation of the radial stress in all models. Given that the chosen node in analysis belongs to the outer wall of the model, as the size of the element becomes more refined and tends towards zero, the value for the radial stress percentual variation tends towards -100% (which means the actual value tends towards zero). It is also possible to see no dependency of the variation of the tangential stress in the size of the radial length of the element.

4.1.2. MATLAB & Abaqus Comparison

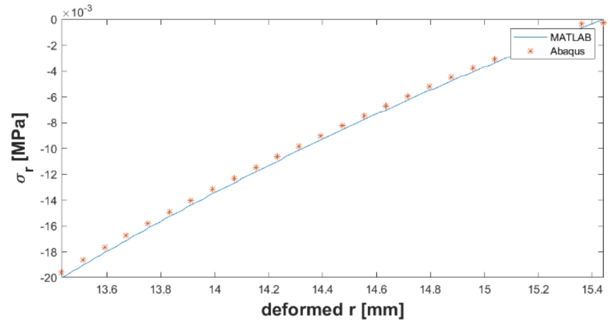

Figure 23 - Comparison between radial stress for the MATLAB and Abaqus WSE1 for a certain IVS input

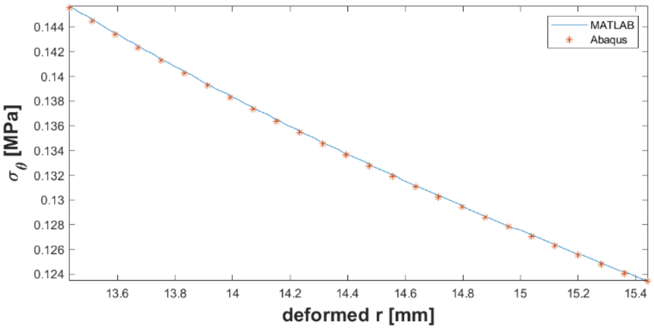

Figure 24 - Comparison between tangential stress for the MATLAB and Abaqus WSE1 for a certain IVS input

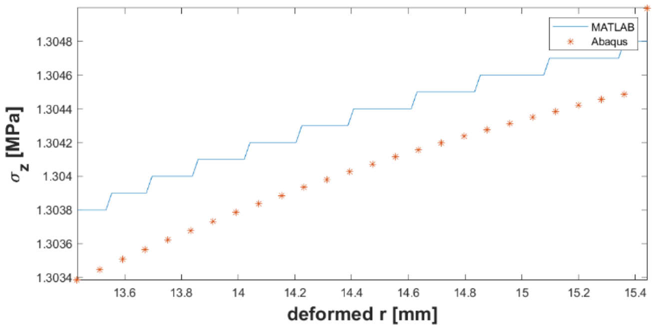

Figure 25 - Comparison between axial stress for the MATLAB and Abaqus WSE1 for a certain input

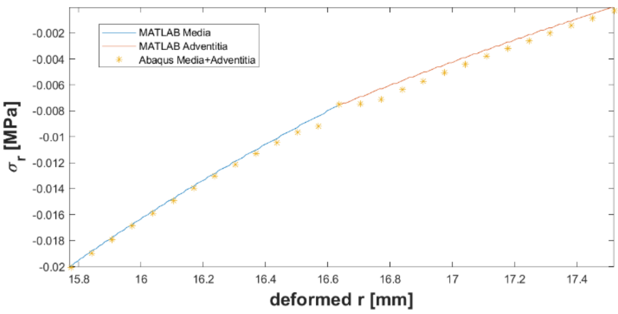

Figure 26 - Comparison between radial stress for the MATLAB and Abaqus WSE2 for a certain IVS input

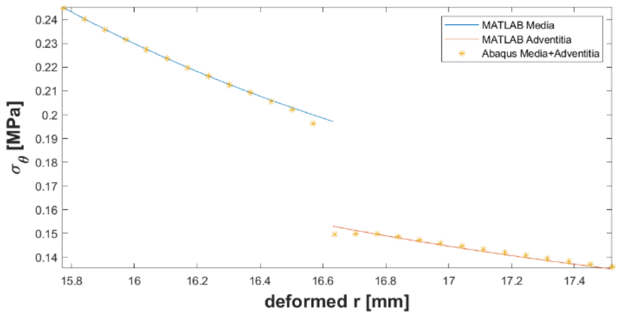

Figure 27 - Comparison between tangential stress for the MATLAB and Abaqus WSE2 for a certain IVS input

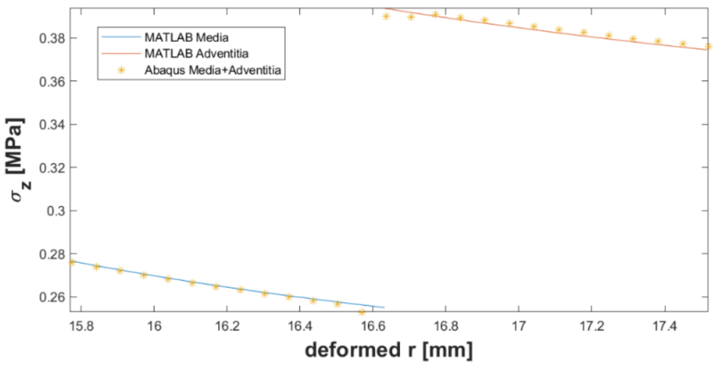

Figure 28 - Comparison between axial stress for the MATLAB and Abaqus WSE2 for a certain IVS input

Table 14 - CPU and Elapsed times (CPUT and ET) for example inputs in WSE1

	CPUT [s]	ET [s]
MATLAB	1.8438	1.7996
Abaqus	2.0625	56.4889

Table 15 - CPU and Elapsed times (CPUT and ET) for example inputs in WSE2

	CPUT [s]	ET [s]
MATLAB	3.7656	3.6976
Abaqus	2.9531	53.1331

Table 16 - CPU and Elapsed times (CPUT and ET) for example inputs in WSE3

	CPUT [s]	ET [s]
Abaqus	516	449

Since the MATLAB and Abaqus values coincide in Figures 23 to 28, it is safe to assume the scripts developed for both softwares are valid for use (it should be noted that, although the Abaqus and MATLAB values do not coincide in Figure 25, the difference between the two is in the order of 0.0001 MPa, which is negligible). Moreover, the much higher elapsed times for the Abaqus simulations of WSE1 and WSE2 (Tables 14 and 15) serve as justification to use MATLAB as the main formulation for sampling the datasets.

4.2. Design of Experiments

4.2.1. Sample Rejection in WSE1 & WSE2

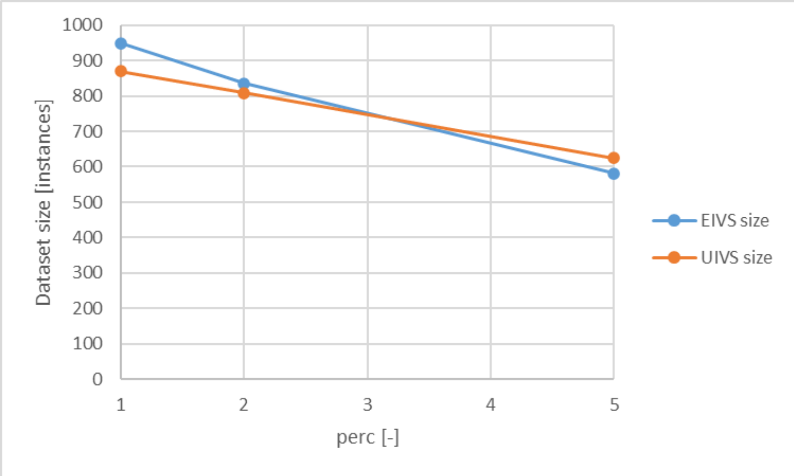

Figure 29 - Number of samples left in the WSE1 EIVS and UIVS after rejection for different $perc$

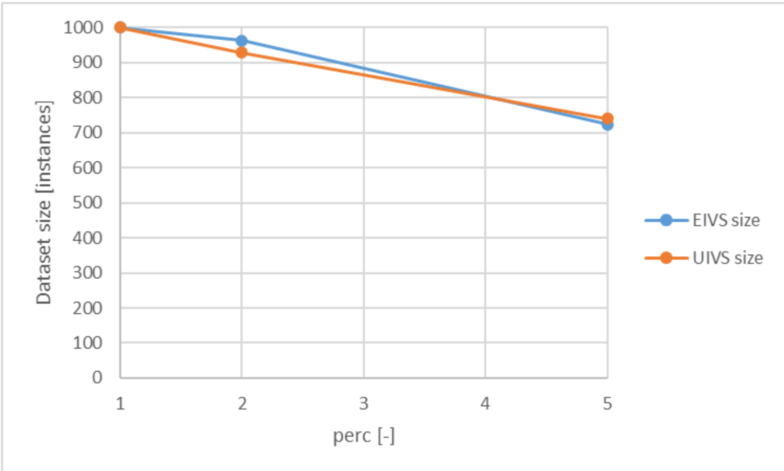

Figure 30 - Number of samples left in the WSE1 EIVS and UIVS after rejection for different $perc$

Figures 29 and 30 indicate an expected decrease in obtained samples as one increases the $perc$ parameter (the ideal number of samples, without rejection, would be 1000, as mentioned previously). It is interesting to note the higher number of rejections in the WSE1 model compared to the WSE2.

4.2.2. *Output Statistics*

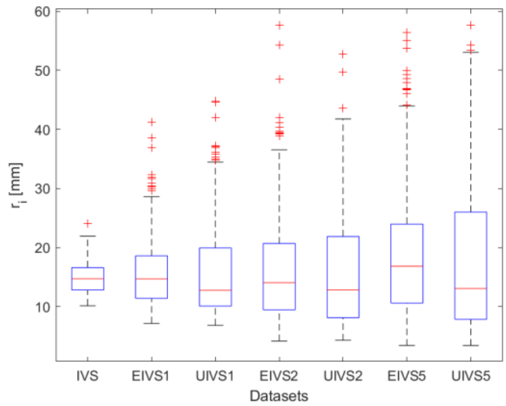

Figure 31 - Boxplots of the WSE1 r_i for the different generated datasets

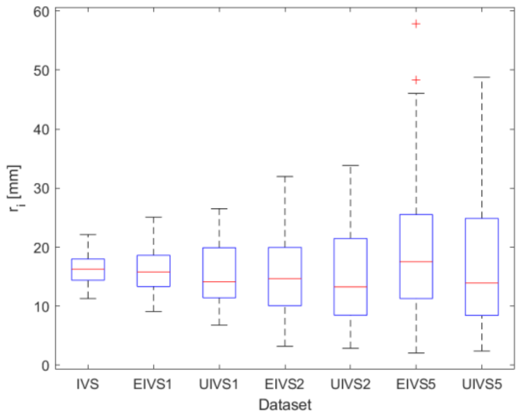

Figure 32 - Boxplots of the WSE2 r_i for the different generated datasets

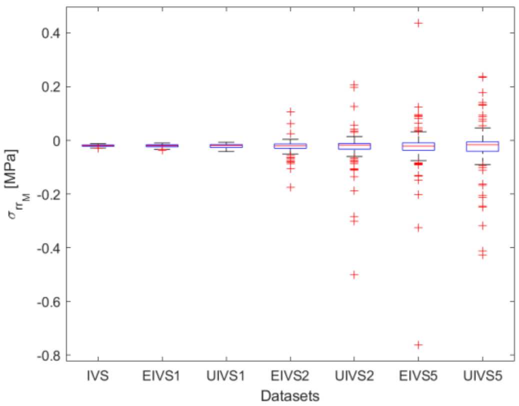

Figure 33 - Boxplots of the WSE2 ${\sigma_{rr}}_M$ for the different generated datasets

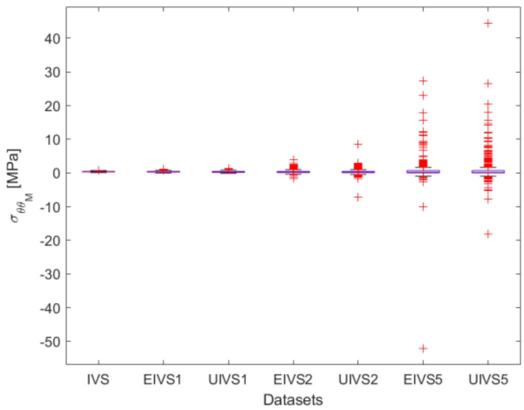

Figure 34 - Boxplots of the WSE2 ${\sigma_{\theta\theta}}_M$ for the different generated datasets

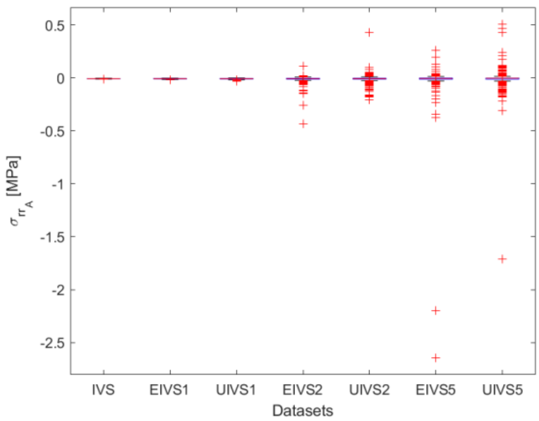

Figure 35 - Boxplots of the WSE2 ${\sigma_{rr}}_A$ for the different generated datasets

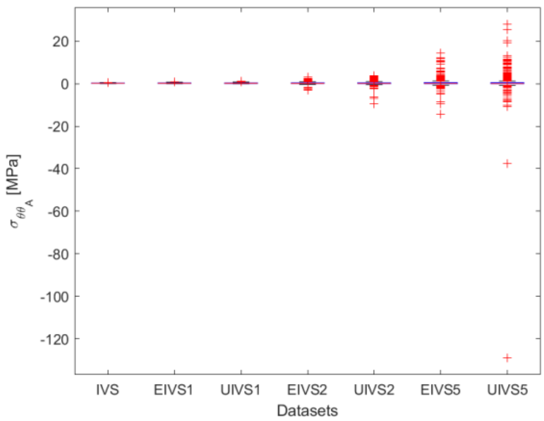

Figure 36 - Boxplots of the WSE2 ${\sigma_{\theta\theta}}_A$ for the different generated datasets

The figures above show a clear increase in the number of outliers (represented by the red markers) and in the spread of the quartiles for all outputs as the $perc$ increases. It should be mentioned that the stress outputs of WSE1 were not considered since no surrogate was built for these outputs.

4.3. Surrogate Models

4.3.1. WSE1

4.3.1.1. WSE1SM1

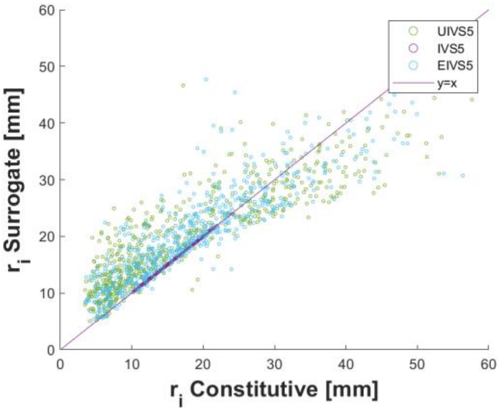

Figure 37 - Predicted VS Real plot for WSE1SM1 for $perc$ = 5

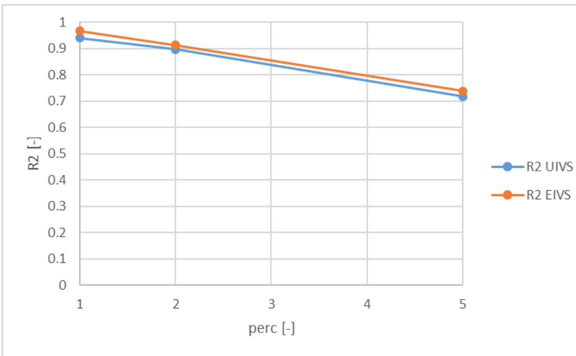

Figure 38 - R^2 VS perc for the UIVS and EIVS ($R^2 = 1.00$ for IVS)

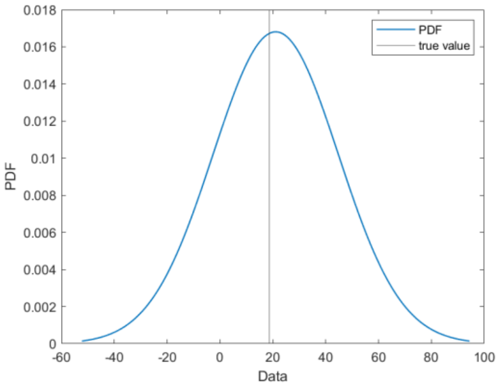

Figure 39 - PDF for an input set from dataset UIVS5

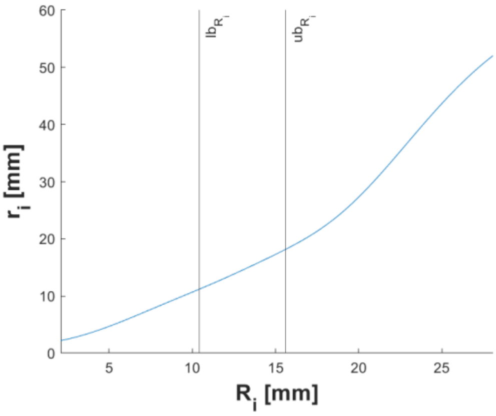

Figure 40 - Partial Dependence Plot of r_i on R_i, with the remaining parameters being constant and equal to their $refval$

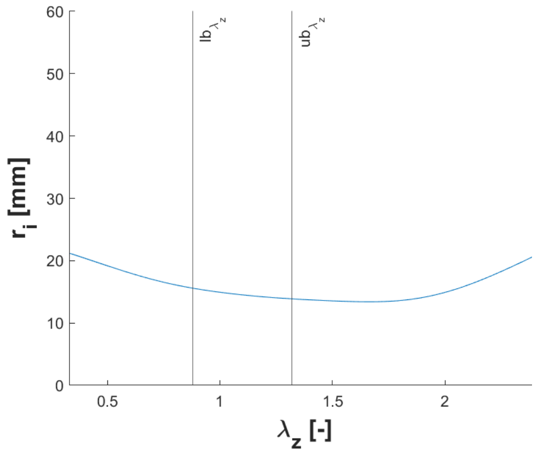

Figure 41 - Partial Dependence Plot of r_i on λ_{zz}, with the remaining parameters being constant and equal to their $refval$

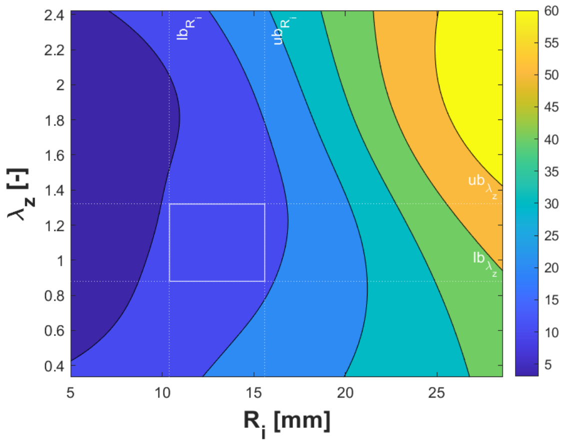

Figure 42 - Partial Dependence Plot of r_i on R_i and λ_{zz}, with the remaining parameters being constant and equal to their refval

Table 17 – Hyperparameters for WSE1SM1

Hyperparameters	Value
Type	GPR
Output	r_i
Basis function	zero
Kernel function	Isotropic rational quadratic
Kernel scale	3.7661
Sigma	0.0073
Standardization	yes

Figure 37 shows that WSE1SM1 is able replicate the outputs fairly well for the testing IVS, and more or less so for the UIVS and EIVS, but it does overestimate the deformed inner radius value for lower real values of the CM, and it underestimates the deformed inner radius value for higher real values of the CM. Given that the coefficient of determination is higher than 0.7/0.8, one can say that this model is a suitable surrogate (Figure 38). The stochastic nature of the GPR allows for the quantification of the uncertainty of the inner deformed radius for a given input set (Figure 39). Figures 40, 41 and 42 show the variation of the inner deformed radius when one varies the undeformed inner radius, the axial stretch or both, respectively, while keeping the remaining inputs constant (the white square represents the projection of IVS into that two-dimensional plane).

4.3.1.2. WSE1SM2

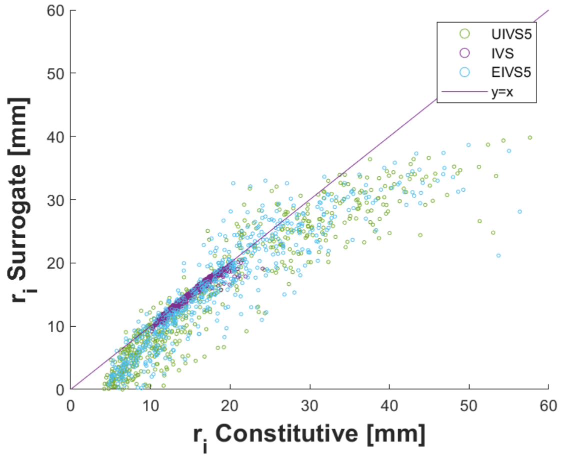

Figure 43 - Predicted VS Real plot for WSE1SM2 for $perc$ = 5

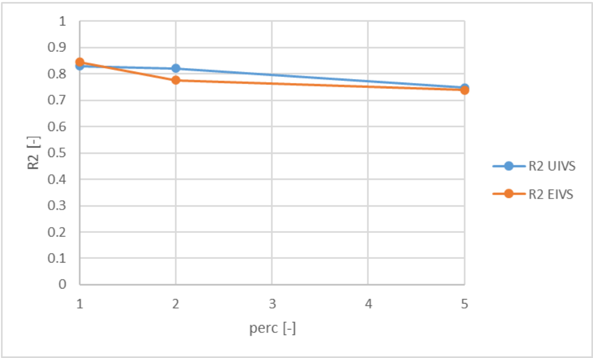

Figure 44 - R^2 VS $perc$ for the UIVS and EIVS ($R^2 = 0.9586$ for IVS)

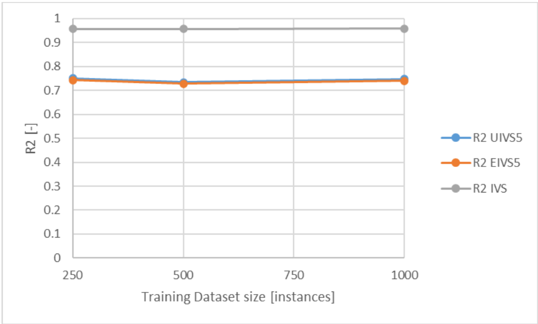

Figure 45 - R^2 VS Training dataset size for IVS, UIVS and EIVS for $perc = 5$

Table 18 - Hyperparameters of WSE1SM2

Hyperparameter	Value
Type	NN
Output	r_i
Number of layers	3
Activation function	none
Layer sizes	[85 19 1]
Regularization	1.677e-7
Standardization	yes

Similarly to the previous model, the R^2 values are above 0.7 for all testing datasets (Figure 44). However, Figure 43 shows an arch shape in the scatter plot for $perc = 5$ (underestimation of the output when the real value is further away from the IVS region in purple). It should also be noted that there is no dependency on the performance of the NN with the size of the training dataset (Figure 45).

4.3.2. WSE2

4.3.2.1. <u>WSE2SM1</u>

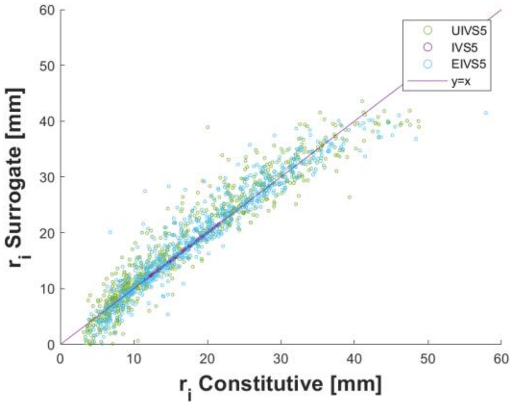

Figure 46 - Predicted VS Real plot for WSE2SM1 for $perc$ = 5

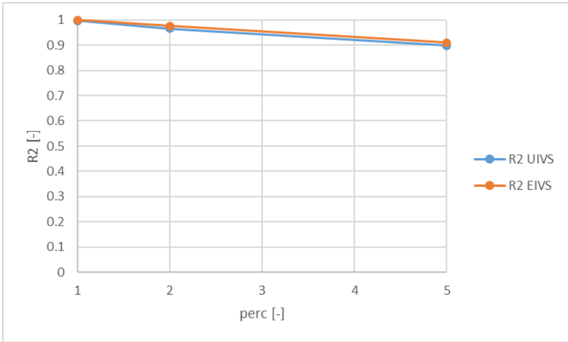

Figure 47 - R^2 VS perc for the UIVS and EIVS (R^2 IVS=0.9999)

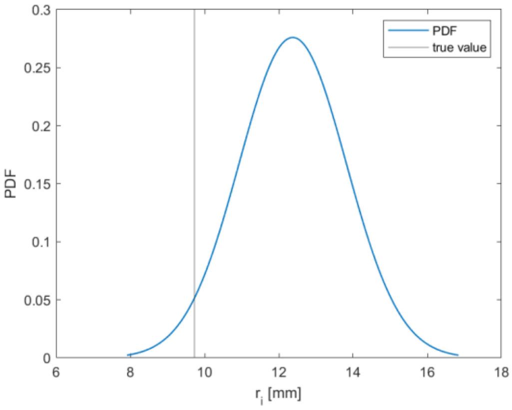

Figure 48 - PDF for an input set from the dataset UIVS5

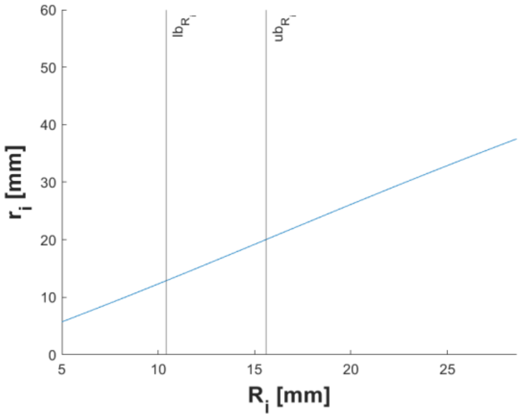

Figure 49 - Partial Dependence Plot of r_i on R_i, with the remaining parameters being constant and equal to their $refval$

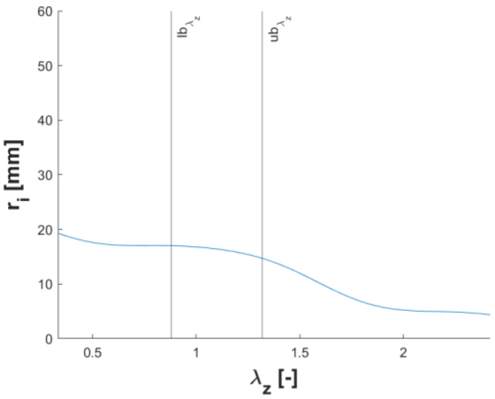

Figure 50 - Partial Dependence Plot of r_i on λ_{zz}, with the remaining parameters being constant and equal to their $refval$

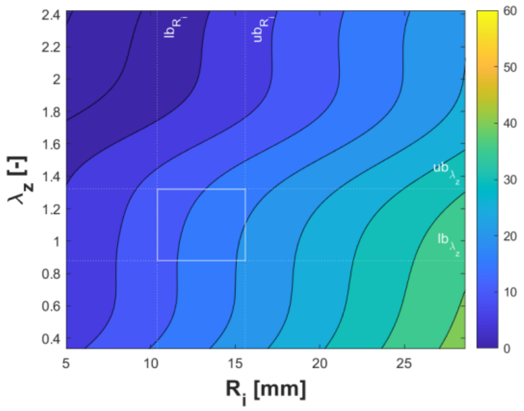

Figure 51 - Partial Dependence Plot of r_i on R_i and λ_{zz}, with the remaining parameters being constant and equal to their $refval$

Table 19 - Hyperparameters of WSE2SM1

Hyperparameter	**Value**
Type	GPR
Output	r_i
Basis function	linear
Kernel function	Nonisotropic squared exponential
Kernel scale	0.95556
Sigma	0.0026056
Standardization	no

Compared to the previous GPR, this surrogate performs a lot better, and the predicted VS real relation is near linear (Figure 46 and 47). Once again, the output's uncertainty is quantified by the PDF for each given input (Figure 48). When it comes to the dependency plots, the relationships between these inputs seem to be similar as the ones depicted by WSE1SM1, however with not as high values for the inner deformed radius (Figures 49-51).

4.3.2.2. WSE2SM2

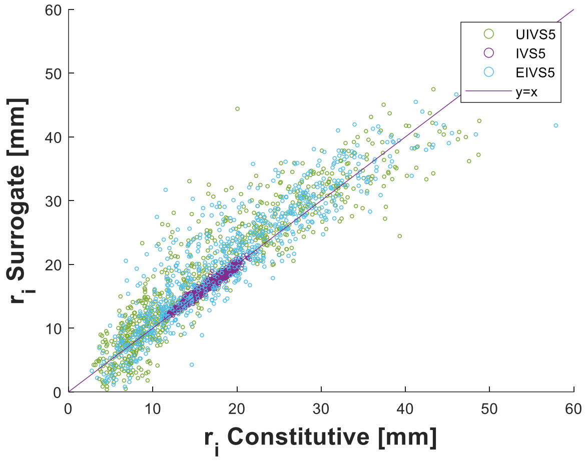

Figure 52 - Predicted VS Real plot for WSE2SM2 for $perc$ = 5

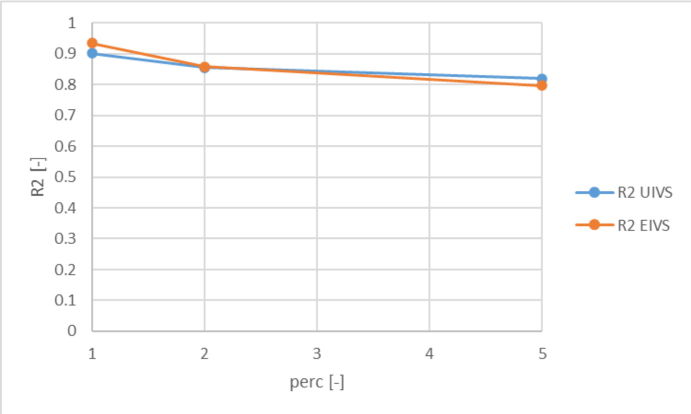

Figure 53 - R^2 VS perc for the UIVS and EIVS (R^2 IVS=0.9614)

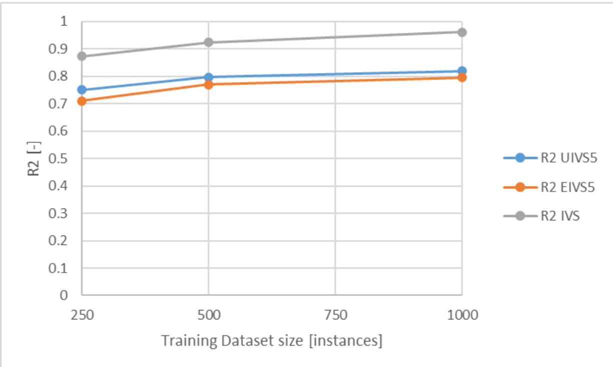

Figure 54 - Study on the influence of the training set size on R^2: R^2VS Training dataset size for IVS, UIVS and EIVS for $perc = 5$

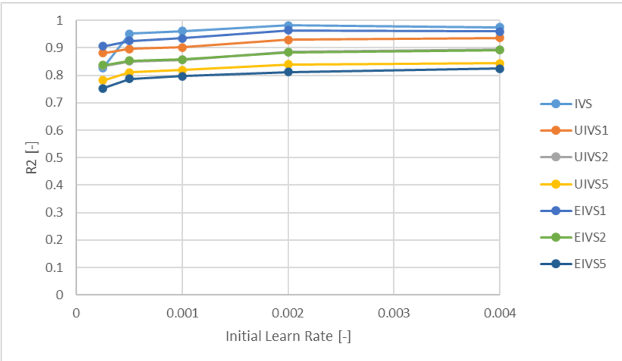

Figure 55 - Sensitivity of R^2 towards the Initial Learn Rate

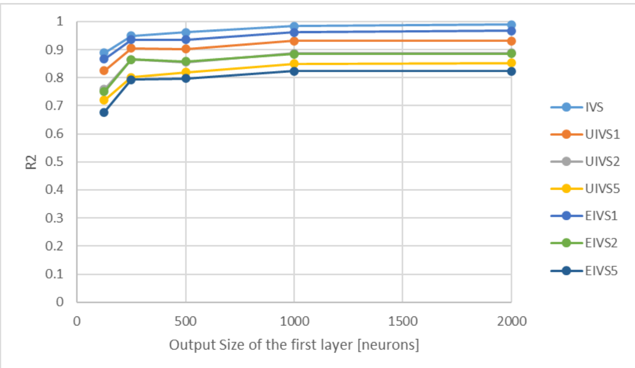

Figure 56 - Sensitivity of R^2 towards the Output Size of the first layer

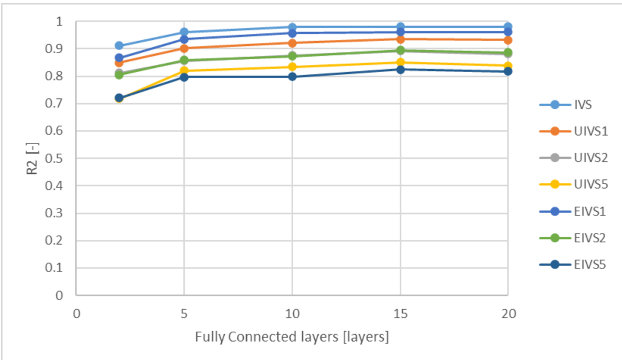

Figure 57 - Sensitivity of R^2 towards the Fully Connected layers

Table 20 - Hyperparameters of WSE1SM2

Hyperparameter	Value
Type	NN
Output	r_i
Number of layers	5
Activation function	ReLu
Layer sizes	[500 375 251 126 1]
L2 Regularization	0.0001
Epochs	10
Batch size	32
Initial learn rate	0.001
Learn rate drop factor	0.5
Learn rate drop period	2
Standardization	yes, for inputs (symmetric)

Comparing Figures 43 and 52, it also seems the neural network for WSE2 possesses higher accuracy and better depiction of the output values. Figure 54-57 consist in a hyperparameter sensitivity study, that show a stagnation of the R^2 after a given value for each parameter; additionally, lower values for number of neurons of the first layer cause sharper decrease in accuracy compared to lower values of the other hyperparameters.

4.3.2.3. WSE2SM3

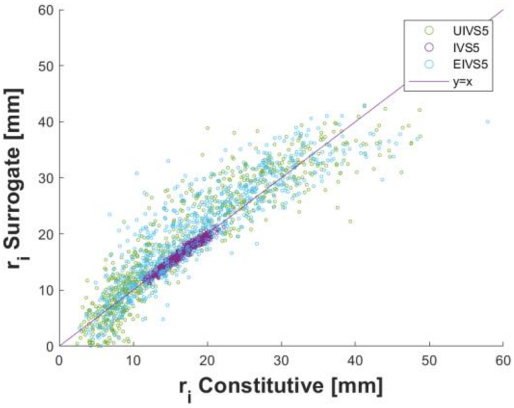

Figure 58 - r_i Predicted VS Real plot for WSE2SM3 for $perc$ = 5

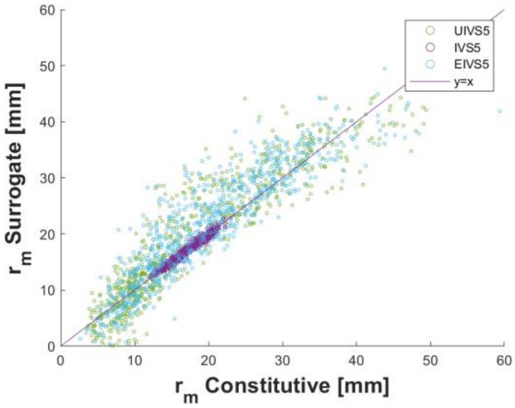

Figure 59 - r_m Predicted VS Real plot for WSE2SM3 for $perc$ = 5

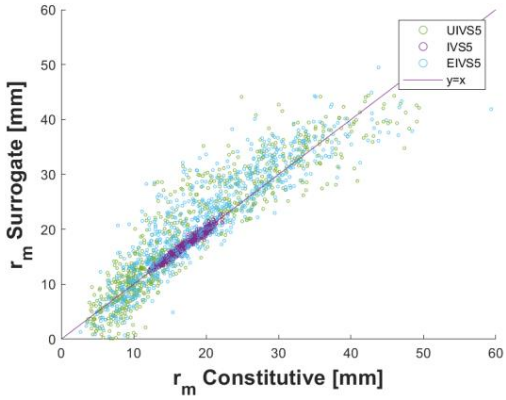

Figure 60 - r_o Predicted VS Real plot for WSE2SM3 for $perc$ = 5

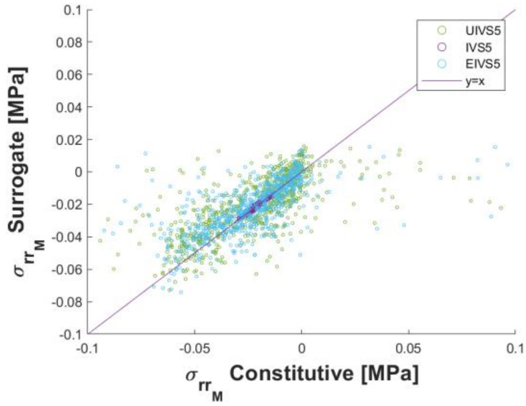

Figure 61 - σ_{rr_M} Predicted VS Real plot for WSE2SM3 for $perc$ = 5

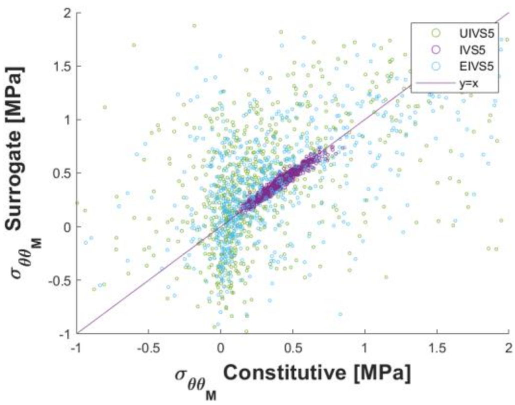

Figure 62 - $\sigma_{\theta\theta_M}$ Predicted VS Real plot for WSE2SM3 for $perc$ = 5

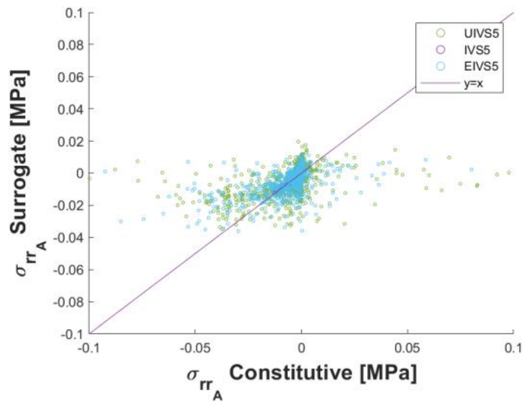

Figure 63 - σ_{rr_A} Predicted VS Real plot for WSE2SM3 for $perc$ = 5

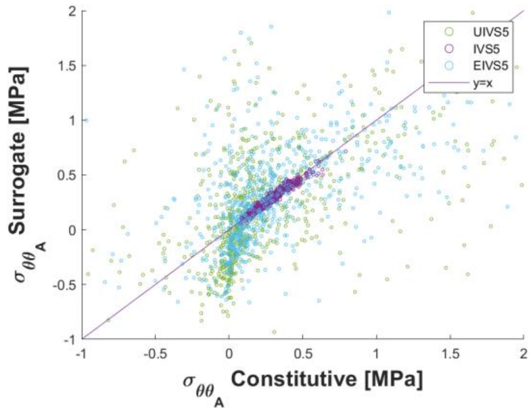

Figure 64 - $\sigma_{\theta\theta_A}$ Predicted VS Real plot for WSE2SM3 for $perc$ = 5

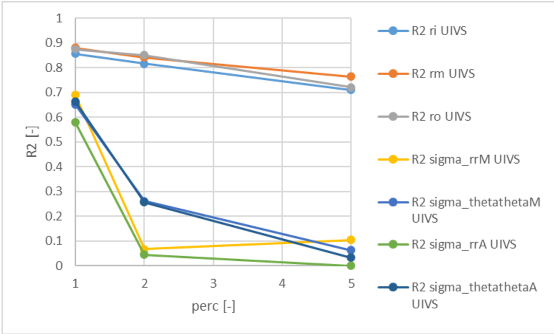

Figure 65 - R^2 VS perc for the UIVS

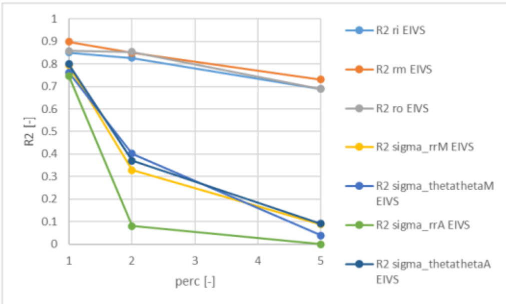

Figure 66 - R^2 VS perc for the EIVS

Table 21 - Hyperparameters of WSE1SM3

Hyperparameters	Value
Type	NN
Output	r_i, r_m, r_o, σ_{rr_M}, $\sigma_{\theta\theta_M}$, σ_{rr_A}, σ_{rr_M}
Remaining Hyperparameters	Same as WSE2SM2

The important aspect of the WSE2SM3 is its attempt at emulating the values of the stress tensor for WSE2. When it comes to the prediction of the deformed radii (Figures 58-60), the surrogate performs well (R^2 above 0.7), while the stress outputs are very badly depicted by the model (R^2 below 0.4 for the UIVS and EIVS). From viewing Figures 61-64, it is possible to see that the predicted radial stresses cluster around the IVS region, while the tangential stresses are a lot more dispersed in their prediction.

4.3.2.4. WSE2SM4

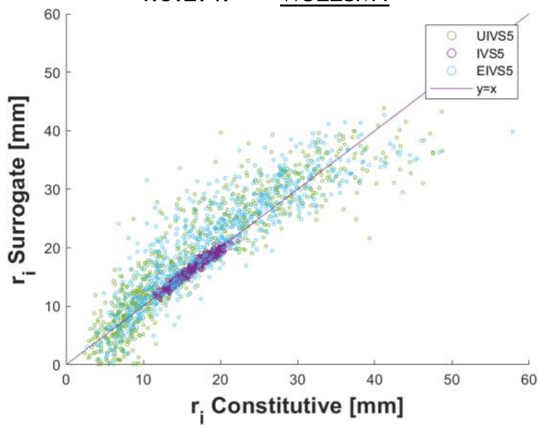

Figure 67 - r_i Predicted VS Real plot for WSE2SM3 for $perc$ = 5

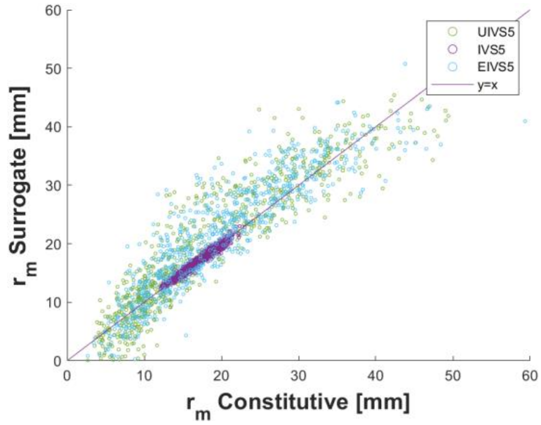

Figure 68 - r_m Predicted VS Real plot for WSE2SM3 for $perc$ = 5

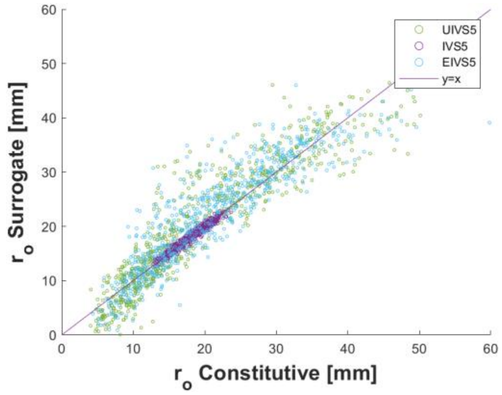

Figure 69 - r_o Predicted VS Real plot for WSE2SM3 for $perc$ = 5

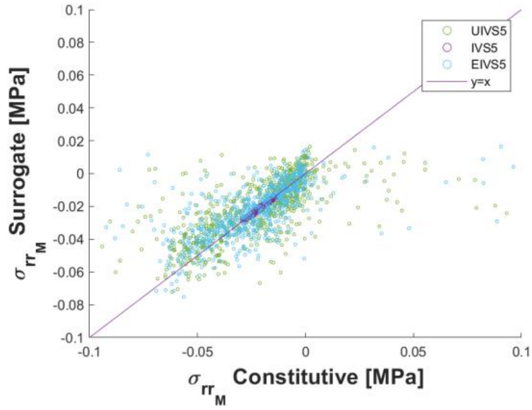

Figure 70 - σ_{rr_M} Predicted VS Real plot for WSE2SM3 for $perc$ = 5

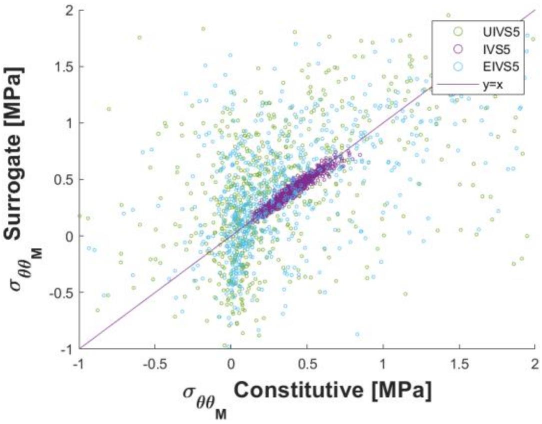

Figure 71 - $\sigma_{\theta\theta_M}$ Predicted VS Real plot for WSE2SM3 for $perc$ = 5

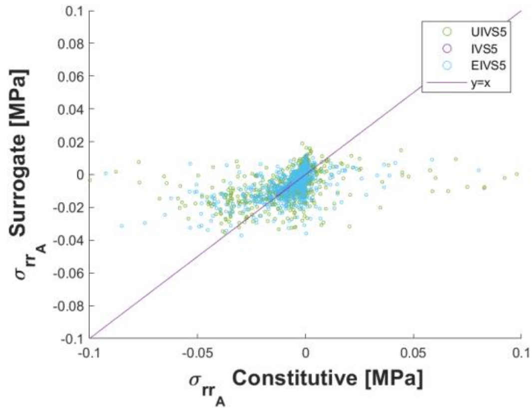

Figure 72 - σ_{rr_A} Predicted VS Real plot for WSE2SM3 for $perc$ = 5

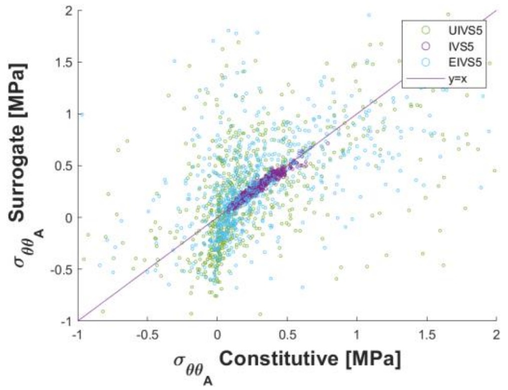

Figure 73 - $\sigma_{\theta\theta_A}$ Predicted VS Real plot for WSE2SM3 for $perc$ = 5

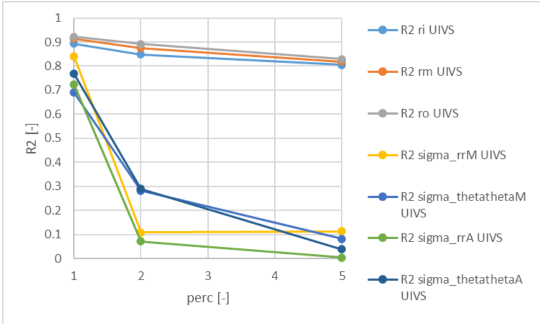

Figure 74 - R^2 VS perc for the UIVS

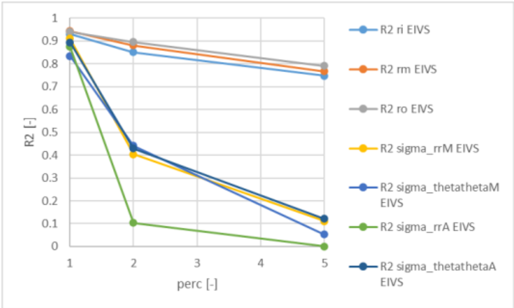

Figure 75 - R^2 VS perc for the EIVS

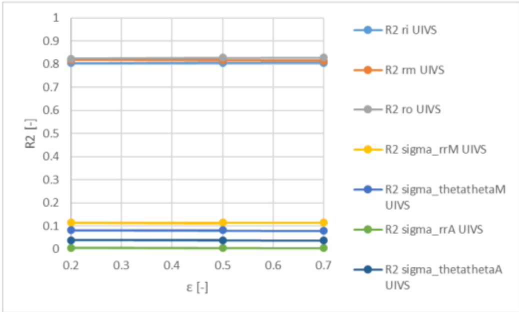

Figure 76 - R^2 VS ε for the UIVS

Table 22 - Hyperparameters of WSE1SM4

Hyperparameters	Value
Type	PINN
Output	r_i, r_m, r_o, σ_{rr_M}, $\sigma_{\theta\theta_M}$, σ_{rr_A}, σ_{rr_M}
Remaining Hyperparameters	Same as WSE2SM2

For WSE2SM4, the objective was to implement physics equations into the loss function of the NN; however, from the analysis of Figures 67-75, it is safe to assume that there is no difference in performance to the WSE2SM3, which does not include physics. Additionally, from Figure 76, no impact on the accuracy is observed by varying the weight of the physics component in the loss function.

4.3.2.5. <u>WSE2SM5</u>

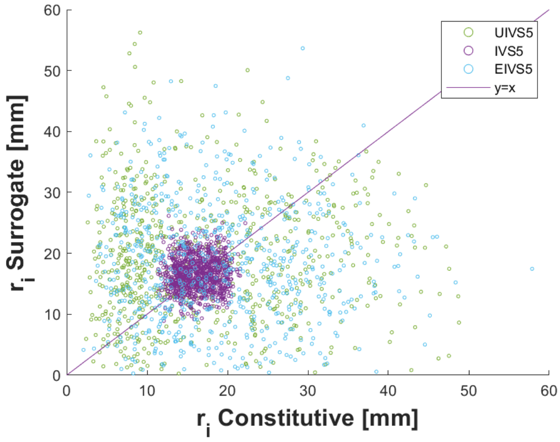

Figure 77 - Predicted VS Real plot for WSE2SM5 for $perc$ *= 5*

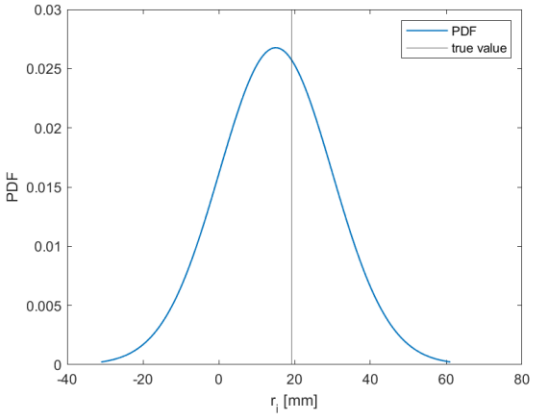

Figure 78 - PDF for an input set from dataset UIVS5

Table 23 - Hyperparameters dor WSE2SM5

Hyperparameter	Value
Type	VIBNN
Output	r_i
Number of layers	2
Activation function	ReLu
Layer sizes	[500 1]
Epochs	50
Batch size	32
Standardization	yes

WSE2SM5 consists of a VIBNN, meaning it possesses the same uncertainty quantification capabilities of a GPR. Regarding the R^2, it possesses a null value for all testing datasets, which is explained by Figure 77, which depicts scattered points that do not cluster around the $y = x$ line. However, it is interesting to note the way the plotted points cluster into squares (a smaller one for the IVS and bigger ones for the EIVS and UIVS testing datasets).

4.3.3. WSE3

4.3.3.1. WSE3SM1

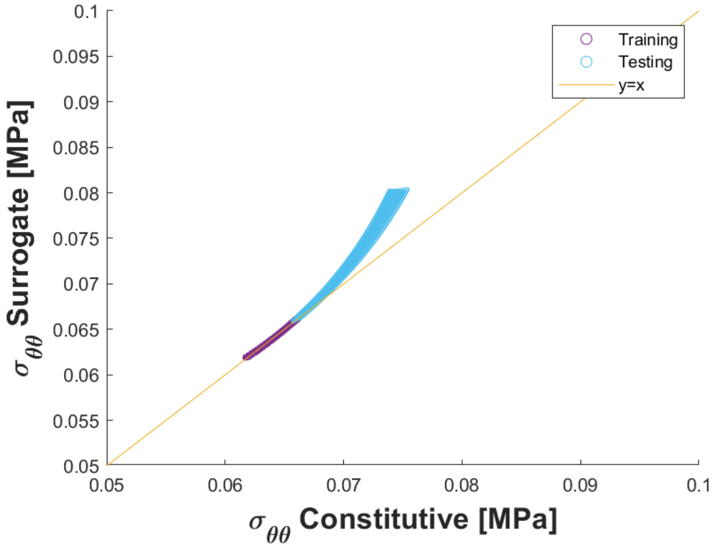

Figure 79 - Predicted VS Real plot for WSE3SM1

Table 24 - Hyperparameters for WSE3SM1

Hyperparameters	Value
Type	GPR
Output	$\sigma_{\theta\theta M}$
Basis function	zero
Kernel function	Nonisotropic Matern 1/2
Kernel scale	9.6857
Sigma	0.0036076
Standardization	yes

The final surrogate, WSESM1, serves to depict the relation between the tangential stress in the arterial wall and the inputs, time (days occurring after the start of growth and remodeling of the fibers) and space (radial point where the stress is being evaluated). Figure 79 depicts an exponential/overestimation relation between the real and predicted values for the testing dataset.

5. Discussion

Firstly, the mesh size convergence tests (Figures 20-22) exhibit the expected response from a formulation that assumes the radial stress at the outer extremity as null. Although a study regarding the size of the elements in the tangential direction was created, it was not presented in the report since there was practically no variation in the tangential or radial stresses in that regard, which is expected given the axisymmetric formulation of the models.

By comparing the MATLAB and Abaqus stress outputs (Figures 23-28), it was possible to validate to cross validate the developed MATLAB code and the Abaqus model. However, one needs to keep in mind that the figures in question are for singular input sets, belonging to IVS. When computing the outputs for certain inputs from EIVS or UIVS, peculiar scenarios surged: boundary conditions not being fulfilled by Abaqus, a complete incompatibility between Abaqus and MATLAB, or similar deformations between the two models but very different stresses. The justification for these occurrences might lie in the extreme nature of some of the input parameters (layers of the arterial wall lower than 0.1 mm, internal wall pressures higher than 0.1 MPa, etc.) which cause the Abaqus and MATLAB simulations to numerically reach different solutions.

This aspect was the motive for the implementation of the rejection of certain samples, which is depicted in subsection 3.2.3. From Figures 29 and 30, as the value of $perc$ increases (the EIVS and UIVS increase in size), more unusual variable combinations (when compared to values one could hypothetically measure from experimental setups) get sampled, and thus do not respect the rejection criteria.

In subsection 4.2.2, the boxplots provide us with a good insight into the distribution of the model outputs. It is interesting to note that, regarding r_i, WSE1, the simpler model, possesses many more outliers than WSE2, which is more complex (Figures 31 and 32). This fact is intertwined, as well, with the fact that there was a higher number of rejections when sampling from WSE1 (Figure 29) compared to WSE2 (Figure 30). On another note, one should highlight the positive values shown in Figures 33 and 35 for the radial stress; these scenarios arise when the radial stress functions (media and adventitia) are not monotonic; i.e., the radial stress in the media rises from $-p_i$ to a certain positive value, from which the radial stress in the adventitia descends to the value of 0, preserving the boundary conditions of the formulations. The last note on this aspect should be towards the values of the stresses in the adventitia (Figures 35 and 36), which seem more likely to have extreme outliers than the media stresses (Figures 33 and 34).

Moving on to the surrogate models, the R^2 VS $perc$ plots show us that, as expected, the R^2 metric decreases with the increased size of the testing dataset, due to outliers that bypassed the rejection criteria, but also due to the inherent difficulty of predicting unseen inputs.

Though there is no significant difference in performance with decreasing sizes of the training set (Figure 54), throughout this work, it was possible to infer that the quality of the training dataset in its ability to capture different outputs in for the IVS greatly affected the R^2 performance metric (this includes the removal of outliers, the type of sampling method, the probability distribution of the input variables, etc.).

When it comes to the Uncertainty Quantification of these models, the PDF examples serve to quantify the aleatoric uncertainty associated with the computed r_i.

Regarding the partial dependence plots, only the influence of the variables R_i and λ_{zz} was presented, since all other dependence plots were mostly horizontal (meaning the surrogate detected no influence on the inner deformed radius when all other inputs remained constant). For both WSE1SM1 and WSE2SM1, there is a positive and negative proportional influence on the r_i by R_i and λ_{zz}, respectively (Figures 40, 41, 49, 50); this is possibly due to the incompressibility condition which the constitutive models need to obey (equation (44)) and which greatly influences the dataset outputs for the deformed inner radius.

From the dual dependence plots (Figures 42 and 51), it is possible to see the contradicting information provided by the GPRs. The contour region inside the IVS space (white square) seems to indicate values in the same value range for the two models, but, outside of it, WSE1SM1 assumes unusually high values for high R_i and high λ_{zz}, while WSE2SM1 indicates high values (although not as high as the ones indicated by WSE1S1) for high R_i and low λ_{zz}.

Moving to the multioutput NN (WSE2SM3), there is a higher difficulty in the NN to predict tangential stresses in comparison to radial stresses, possibly due to the radial stresses being confined, most of the time, to the interval $[-p_i \quad 0]$, while the sampled tangential stresses are scattered across different magnitudes (also see Figures 34 and 36).

When it comes to the implementation of physics in NN (WSE2SM4, Figures 67-73), there is no difference between the PINN and the equivalent non-PI NN (WSE2SM3). On one hand, this may be because the physics loss had to be normalized (using zscore) in order to implement physics, since extremely high loss values were computed during training, making the learning process impossible. On the other hand, it may be also because of the complexity of the governing equation (equation (38)): papers that reference the implementation of PINN [62, 63] normally involve networks with one output that is differentiated (in reference to one or two inputs), and then these derivatives are evaluated in a PDE (Partial differential equation), which is simpler than the task at hand involving differentiation of multiple outputs in reference to an input which is not easy to sample from.

Regarding the VIBNN (Figure 77), although there is no reliable deterministic mapping of the outputs, the network is capable of capturing the expected possible range of r_i values for the different types of datasets, as previously said.

Finally, the GPR for G&R captures an exponential relation between depicted and the real constitutive values of the tangential stress. It should be noted that the different appearance of its Predicted VS Real Values plot in comparison to the previous ones is due to the “sampling” process adopted. Nonetheless, once aware of this relationship, one could adapt the surrogate to produce more accurate values (by introducing a transformation function that removes the exponentiality).

6. Conclusions

With the completion of this project, it was possible to attain a range of skills related to computational biomechanics. Not only it was possible to develop technical competences related to code implementation (such as the creation of MATLAB scripts or Abaqus UMATs), but it was also possible to acquire essential knowledge in computational mechanics, related to modeling frameworks, quantification of uncertainty, and critical observation of obtained results.

Concerning the possible implementation of these models in broader sense, one can infer that there is reliability in the prediction of deformation for arterial wall models similar to the ones represented by WSE1 and WSE2, and thus, there is potential use of these surrogates in computational studies associated with quantification of uncertainty, or global dynamic analysis. When it comes to stress prediction for WSE1 and WSE2, further studies would have to be conducted on how to improve the performance of the surrogates, knowing that the biggest challenge lies in detecting and filter out the high prevalence of outliers in the stress data, and also in the method adopted to sample the stresses at different geometric locations. Looking into the use of the WSE3, although the surrogate is more limited compared to the other ones (in terms of the range of inputs it can accept), there is promising use to predict the tangential stress for longer periods of growth and remodeling that may be unpractical to obtain through the constitutive model, due to processing times.

Regarding future works, one may reference other approaches that should be attempted to map the stresses of arterial walls better. This includes the implementation of other surrogate formats such as tensor-based networks [84-86] or decoders [87], improved design of experiments (namely, including criteria for resampling and not rejection) or a more extensive application of the surrogates in the quantification of uncertainty as mentioned above. Additionally, the constitutive models could be enhanced in their depiction of real-life arterial behaviour, namely by including more realistic geometries, material models (including damage, for example), etc.

Although this project was not always straightforward, and some implementations were not successful, and, therefore, not included in the report, the experience gained in handling various challenges will prove invaluable in possible future ventures within the scope of computational modeling for mechanical engineering.

References

[1] M. Kaushik and R. Kumar, "A Conceptual Study on Digital Transformation & its Applications: An Overview."

[2] H. Chin, D. P. Marasini, and D. Lee, "Digital transformation trends in service industries," *Service Business,* vol. 17, no. 1, pp. 11-36, 2023.

[3] A. I. Stoumpos, F. Kitsios, and M. A. Talias, "Digital Transformation in Healthcare: Technology Acceptance and Its Applications," *Int J Environ Res Public Health,* vol. 20, no. 4, Feb 15 2023, doi: 10.3390/ijerph20043407.

[4] E. Katsoulakis *et al.*, "Digital twins for health: a scoping review," *NPJ Digit Med,* vol. 7, no. 1, p. 77, Mar 22 2024, doi: 10.1038/s41746-024-01073-0.

[5] "INNOVAHEART – A joint European workshop on the Digital Heart." https://eithealth.eu/news-article/innovaheart-a-joint-european-workshop-on-the-digital-heart/ (accessed 06, 2024).

[6] S. Chakraborty, S. Adhikari, and R. Ganguli, "The role of surrogate models in the development of digital twins of dynamic systems," *Applied Mathematical Modelling,* vol. 90, pp. 662-681, 2021.

[7] L. L. Fonseca, L. Böttcher, B. Mehrad, and R. C. Laubenbacher, "Metamodeling and control of medical digital twins," *arXiv preprint arXiv:2402.05750,* 2024.

[8] M. Smoljkić *et al.*, "Biomechanical characterization of ascending aortic aneurysms," *Biomechanics and modeling in mechanobiology,* vol. 16, pp. 705-720, 2017.

[9] A. M. Taylor and B. Bordoni, "Histology, blood vascular system," 2020.

[10] W. D. Tucker, Y. Arora, and K. Mahajan, "Anatomy, blood vessels," 2017.

[11] J. Gupta and M. Shea, "Biology of the Blood Vessels," *Merck Manuals Consumer Version,* 2019.

[12] "Vascular Disease (Vasculopathy)." https://my.clevelandclinic.org/health/diseases/17604-vascular-disease (accessed 06, 2024).

[13] E. S. Wise, K. M. Hocking, and C. M. Brophy, "Prediction of in-hospital mortality after ruptured abdominal aortic aneurysm repair using an artificial neural network," *Journal of vascular surgery,* vol. 62, no. 1, pp. 8-15, 2015.

[14] D. Yamanouchi, "Unpacking the Complexities of a Silent Killer," vol. 24, ed: MDPI, 2023, p. 7125.

[15] T. A. Nezwek and M. Varacallo, "Physiology, Connective Tissue," 2019.

[16] B. B. Hafen and B. Burns, "Physiology, smooth muscle," 2018.

[17] "artery." https://www.britannica.com/science/artery (accessed 06, 2024).

[18] A. A. Mercadante and A. Raja, "Anatomy, arteries," 2019.

[19] J.-L. Gade, *Mechanical properties of arteries: Identification and Application*. Linköping University Electronic Press, 2019.

[20] T. C. Gasser, R. W. Ogden, and G. A. Holzapfel, "Hyperelastic modelling of arterial layers with distributed collagen fibre orientations," *Journal of the royal society interface,* vol. 3, no. 6, pp. 15-35, 2006.

[21] H. W. Weizsacker and J. G. Pinto, "Isotropy and anisotropy of the arterial wall," *Journal of biomechanics,* vol. 21, no. 6, pp. 477-487, 1988.

[22] K. Rhee and Y. Cho, "Artery wall viscoelasticity: measurement, assessment, and clinical implications," *International Journal of Precision Engineering and Manufacturing,* vol. 22, no. 6, pp. 1157-1168, 2021.

[23] B. Van Varik, R. Rennenberg, C. Reutelingsperger, A. Kroon, P. de Leeuw, and L. Schurgers, "Mechanisms of arterial remodeling: lessons from genetic diseases. Front Genet. 2012; 3: 290," ed: PUBMED, 2012.

[24] N. F. Renna, N. De Las Heras, and R. M. Miatello, "Pathophysiology of vascular remodeling in hypertension," *International journal of hypertension,* vol. 2013, no. 1, p. 808353, 2013.

[25] A. Sarvazyan, A. Tatarinov, and N. Sarvazyan, "Ultrasonic assessment of tissue hydration status," *Ultrasonics,* vol. 43, no. 8, pp. 661-671, 2005.
[26] R. Pahwa and I. Jialal, "Atherosclerosis," 2023.
[27] S. Sethi and U. Parekh, "Aortic Arch Aneurysm," in *StatPearls [Internet]*: StatPearls Publishing, 2023.
[28] E. Saliba, Y. Sia, A. Dore, and I. El Hamamsy, "The ascending aortic aneurysm: When to intervene?," *IJC Heart & Vasculature,* vol. 6, pp. 91-100, 2015.
[29] R. M. Hackett, *Hyperelasticity primer*. Springer, 2016.
[30] Y.-c. Fung, *Biomechanics: mechanical properties of living tissues*. Springer Science & Business Media, 2013.
[31] J. Bonet and R. D. Wood, *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press, 1997.
[32] G. A. Holzapfel, T. C. Gasser, and R. W. Ogden, "A new constitutive framework for arterial wall mechanics and a comparative study of material models," *Journal of elasticity and the physical science of solids,* vol. 61, pp. 1-48, 2000.
[33] C. J. Cyron, R. Aydin, and J. D. Humphrey, "A homogenized constrained mixture (and mechanical analog) model for growth and remodeling of soft tissue," *Biomechanics and modeling in mechanobiology,* vol. 15, pp. 1389-1403, 2016.
[34] J. D. Humphrey and K. Rajagopal, "A constrained mixture model for growth and remodeling of soft tissues," *Mathematical models and methods in applied sciences,* vol. 12, no. 03, pp. 407-430, 2002.
[35] L. Maes and N. Famaey, "How to implement constrained mixture growth and remodeling algorithms for soft biological tissues," *Journal of the Mechanical Behavior of Biomedical Materials,* vol. 140, p. 105733, 2023.
[36] F. A. Bräu, A. Seitz, R. Aydin, and C. J. Cyron, "Homogenized constrained mixture models for anisotropic volumetric growth and remodeling," *Biomechanics and modeling in mechanobiology,* vol. 16, pp. 889-906, 2017.
[37] A. M. Gebauer, M. R. Pfaller, F. A. Braeu, C. J. Cyron, and W. A. Wall, "A homogenized constrained mixture model of cardiac growth and remodeling: Analyzing mechanobiological stability and reversal," *Biomechanics and modeling in mechanobiology,* vol. 22, no. 6, pp. 1983-2002, 2023.
[38] F. Sempértegui and S. Avril, "Re-interpretation of the Homogenized Constrained Mixture Theory within the plasticity framework and application to soft tissue growth and remodeling," *Computer Methods in Applied Mechanics and Engineering,* vol. 412, p. 116059, 2023.
[39] G. A. Holzapfel and R. W. Ogden, *Biomechanics: trends in modeling and simulation*. Springer, 2017.
[40] N. Famaey, H. Fehervary, L. Maes, K. V. Linden, and J. Vastmans, "Wall Stress Estimator," KU Leuven.
[41] A. F. Bower, *Applied mechanics of solids*. CRC press, 2009.
[42] H. Fehervary, L. Maes, J. Vastmans, G. Kloosterman, and N. Famaey, "How to implement user-defined fiber-reinforced hyperelastic materials in finite element software," *Journal of the Mechanical Behavior of Biomedical Materials,* vol. 110, p. 103737, 2020.
[43] D. Nolan, C. Lally, and J. McGarry, "Understanding the deformation gradient in Abaqus and key guidelines for anisotropic hyperelastic user material subroutines (UMATs)," *Journal of the Mechanical Behavior of Biomedical Materials,* vol. 126, p. 104940, 2022.
[44] D. Systèmes, *Abaqus Analysis User's Manual 6.11*. 2011.
[45] B. Williams and S. Cremaschi, "Novel tool for selecting surrogate modeling techniques for surface approximation," in *Computer Aided Chemical Engineering*, vol. 50: Elsevier, 2021, pp. 451-456.
[46] C. J. Roy and W. L. Oberkampf, "A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing," *Computer methods in applied mechanics and engineering,* vol. 200, no. 25-28, pp. 2131-2144, 2011.
[47] M. J. Asher, B. F. Croke, A. J. Jakeman, and L. J. Peeters, "A review of surrogate models and their application to groundwater modeling," *Water Resources Research,* vol. 51, no. 8, pp. 5957-5973, 2015.

[48] T. Robinson, M. S. Eldred, K. E. Willcox, and R. Haimes, "Surrogate-based optimization using multifidelity models with variable parameterization and corrected space mapping," *AIAA journal*, vol. 46, no. 11, pp. 2814-2822, 2008.
[49] S. Dutta and A. H. Gandomi, "Design of experiments for uncertainty quantification based on polynomial chaos expansion metamodels," in *Handbook of probabilistic models*: Elsevier, 2020, pp. 369-381.
[50] S. Bi, M. Beer, S. Cogan, and J. Mottershead, "Stochastic model updating with uncertainty quantification: an overview and tutorial," *Mechanical Systems and Signal Processing*, vol. 204, p. 110784, 2023.
[51] K. Burrage, P. Burrage, D. Donovan, and B. Thompson, "Populations of models, experimental designs and coverage of parameter space by Latin hypercube and orthogonal sampling," *Procedia Computer Science*, vol. 51, pp. 1762-1771, 2015.
[52] "Latin hypercube sampling." https://en.wikipedia.org/wiki/Latin_hypercube_sampling (accessed 06, 2024).
[53] C. J. Sallaberry, J. C. Helton, and S. C. Hora, "Extension of Latin hypercube samples with correlated variables," *Reliability Engineering & System Safety*, vol. 93, no. 7, pp. 1047-1059, 2008.
[54] S. A. Youssef, J. K. Paik, Y. S. Kim, M. S. Kim, and F. Cheng, "Probabilistic selection of ship-ship collision scenarios," in *International Conference on Offshore Mechanics and Arctic Engineering*, 2013, vol. 55324: American Society of Mechanical Engineers, p. V02AT02A027.
[55] M. A. Nielsen, *Neural networks and deep learning*. Determination press San Francisco, CA, USA, 2015.
[56] C. C. Aggarwal, *Neural networks and deep learning* (no. 978). Springer, 2018.
[57] G. Shahzadi and A. Soulaïmani, "Deep neural network and polynomial chaos expansion-based surrogate models for sensitivity and uncertainty propagation: An application to a rockfill dam," *Water*, vol. 13, no. 13, p. 1830, 2021.
[58] "What is a Neural Network?" https://www.mathworks.com/discovery/neural-network.html (accessed 06, 2024).
[59] R. Pramoditha. "Overview of a Neural Network's Learning Process." Data Science 365. https://medium.com/data-science-365/overview-of-a-neural-networks-learning-process-61690a502fa (accessed 06, 2024).
[60] "Automatic Differentiation." https://en.wikipedia.org/wiki/Automatic_differentiation (accessed 06, 2024).
[61] J. Brownlee. "Use Weight Regularization to Reduce Overfitting of Deep Learning Models." https://machinelearningmastery.com/weight-regularization-to-reduce-overfitting-of-deep-learning-models/ (accessed 06, 2024).
[62] F. Sahli Costabal, Y. Yang, P. Perdikaris, D. E. Hurtado, and E. Kuhl, "Physics-informed neural networks for cardiac activation mapping," *Frontiers in Physics*, vol. 8, p. 42, 2020.
[63] K. Linka, A. Schäfer, X. Meng, Z. Zou, G. E. Karniadakis, and E. Kuhl, "Bayesian Physics Informed Neural Networks for real-world nonlinear dynamical systems," *Computer Methods in Applied Mechanics and Engineering*, vol. 402, p. 115346, 2022.
[64] Y. Yang and P. Perdikaris, "Adversarial uncertainty quantification in physics-informed neural networks," *Journal of Computational Physics*, vol. 394, pp. 136-152, 2019.
[65] L. V. Jospin, H. Laga, F. Boussaid, W. Buntine, and M. Bennamoun, "Hands-on Bayesian neural networks—A tutorial for deep learning users," *IEEE Computational Intelligence Magazine*, vol. 17, no. 2, pp. 29-48, 2022.
[66] I. Oleksiienko, D. T. Tran, and A. Iosifidis, "Variational neural networks," *Procedia Computer Science*, vol. 222, pp. 104-113, 2023.
[67] M. A. Vega and M. D. Todd, "A variational Bayesian neural network for structural health monitoring and cost-informed decision-making in miter gates," *Structural Health Monitoring*, vol. 21, no. 1, pp. 4-18, 2022.
[68] V. Nemani *et al.*, "Uncertainty quantification in machine learning for engineering design and health prognostics: A tutorial," *Mechanical Systems and Signal Processing*, vol. 205, p. 110796, 2023.
[69] J. Wang, "An intuitive tutorial to Gaussian processes regression," *Computing in Science & Engineering*, 2023.

[70] H. Xu, H. Li, G. Xiang, and X. Zhang, "An improved adaptive surrogate model and application in thermal management system design," *Materials & Design*, vol. 208, p. 109883, 2021.
[71] "Bayesian Linear Regression Using Hamiltonian Monte Carlo." https://www.mathworks.com/help/stats/bayesian-linear-regression-using-hamiltonian-monte-carlo.html (accessed 06, 2024).
[72] "Bayesian Analysis for a Logistic Regression Model." https://nl.mathworks.com/help/stats/bayesian-analysis-for-a-logistic-regression-model.html (accessed 06, 2024).
[73] "Specify Training Options in Custom Training Loop." https://nl.mathworks.com/help/deeplearning/ug/specify-training-options-in-custom-training-loop.html (accessed 06, 2024).
[74] "Solve Partial Differential Equations Using Deep Learning." https://nl.mathworks.com/help/deeplearning/ug/solve-partial-differential-equations-using-deep-learning.html (accessed 06, 2024).
[75] "Train Bayesian Neural Network." https://nl.mathworks.com/help/deeplearning/ug/train-bayesian-neural-network.html (accessed 06, 2024).
[76] "Monitor Custom Training Loop Progress." https://nl.mathworks.com/help/deeplearning/ug/monitor-custom-training-loop-progress.html (accessed 06, 2024).
[77] D. Systèmes, *Abaqus Scripting User's Manual 6.11*. 2012.
[78] N. Famaey *et al.*, "Numerical simulation of arterial remodeling in pulmonary autografts," *ZAMM‐Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 98, no. 12, pp. 2239-2257, 2018.
[79] B. Ciszek, K. Cieślicki, P. Krajewski, and S. K. Piechnik, "Critical pressure for arterial wall rupture in major human cerebral arteries," *Stroke*, vol. 44, no. 11, pp. 3226-3228, 2013.
[80] R. Lorbeer *et al.*, "Reference values of vessel diameters, stenosis prevalence, and arterial variations of the lower limb arteries in a male population sample using contrast-enhanced MR angiography," *PloS one*, vol. 13, no. 6, p. e0197559, 2018.
[81] S. Mishani, H. Belhoul-Fakir, C. Lagat, S. Jansen, B. Evans, and M. Lawrence-Brown, "Stress distribution in the walls of major arteries: implications for atherogenesis," *Quantitative imaging in medicine and surgery*, vol. 11, no. 8, p. 3494, 2021.
[82] A. H. Hoffman *et al.*, "Stiffness properties of adventitia, media, and full thickness human atherosclerotic carotid arteries in the axial and circumferential directions," *Journal of biomechanical engineering*, vol. 139, no. 12, p. 124501, 2017.
[83] B. Pierre, A. Stéphane, L. Susan, and S. Michael, "Mechanical identification of hyperelastic anisotropic properties of mouse carotid arteries," in *Mechanics of Biological Systems and Materials, Volume 2: Proceedings of the 2011 Annual Conference on Experimental and Applied Mechanics*, 2011: Springer, pp. 11-17.
[84] J. N. Fuhg, N. Bouklas, and R. E. Jones, "Learning hyperelastic anisotropy from data via a tensor basis neural network," *Journal of the Mechanics and Physics of Solids*, vol. 168, p. 105022, 2022.
[85] K. Linka, M. Hillgärtner, K. P. Abdolazizi, R. C. Aydin, M. Itskov, and C. J. Cyron, "Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning," *Journal of Computational Physics*, vol. 429, p. 110010, 2021.
[86] L. Linden, D. K. Klein, K. A. Kalina, J. Brummund, O. Weeger, and M. Kästner, "Neural networks meet hyperelasticity: A guide to enforcing physics," *Journal of the Mechanics and Physics of Solids*, vol. 179, p. 105363, 2023.
[87] S. Ranftl *et al.*, "Stochastic modeling of inhomogeneities in the aortic wall and uncertainty quantification using a Bayesian encoder-decoder surrogate," *Computer Methods in Applied Mechanics and Engineering*, vol. 401, p. 115594, 2022.