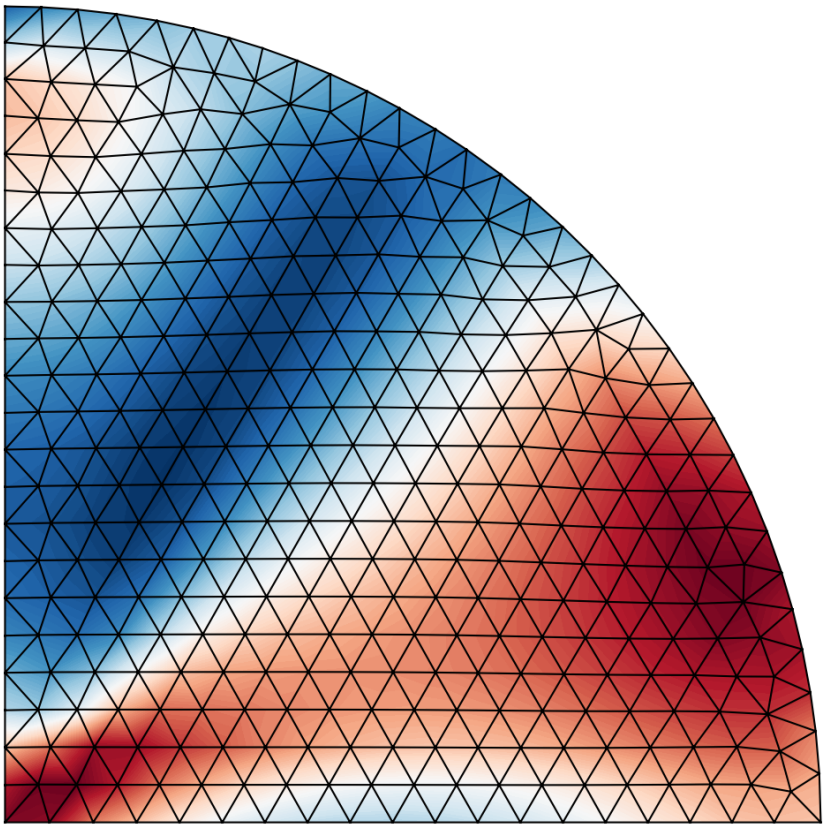
Analysis of Stabilized Finite Element Methods for a Morpho-Poroelastic Model Applied to Tumor Growth

Duncan den Bakker



TU Delft

Analysis of Stabilized Finite Element Methods for a Morpho-Poroelastic Model Applied to Tumor Growth

by

Duncan den Bakker

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday July 27, 2020 at 14:00.

Student number: 4476166
Project duration: December 1, 2020 – July 27, 2020
Thesis committee: Prof. dr. ir. F.J. Vermolen, TU Delft, supervisor
Dr. N.V. Budko, TU Delft
Dr. J.L.A. Dubbeldam, TU Delft

TU Delft

Abstract

The theory of morpho-poroelasticity is applied to tumor growth. This allows for the modeling of permanent deformation in the tissue as a result of the presence of tumor cells. The work done in this project can be divided into two parts. In the first part we review existing models for elasticity and build corresponding finite element models. The aim of the first part is to gain understanding in the behaviour of morphoelasticity and poroelasticity. In morphoelasticity the deformation tensor is decomposed into an elastic and plastic component, allowing for permanent deformations to occur in the tissue. The theory of morphoelasticity describes the interplay between a fluid in a porous tissue and the tissue's elastic properties. A common problem in poroelastic models is the occurence of non-physical oscillations in the pressure. The second part of this project contains novel work. First, a rigorous mathematical derivation is presented of the tuning parameter found in the diffusive stabilization method for poroelastic systems. Secondly, we present a finite element model for the novel combination of morpho- and poroelasticity. The derivation concerning the diffusive stabilization is also applicable to the morpho-poroelastic finite element model. We obtain a promising tumor growth model by combining a biochemcial tumor growth model with morpho-poroelasticity. This biochemical model is based on a nutrient transport equation and a tumor cell density evolution equation. Some example output of the tumor growth finite element model is shown. It can be used as a starting point for more elaborate tumor growth models.

Contents

List of Symbols

In this work we develop two-dimensional models. These models are assumed slices of three-dimensional objects in which there is no change in the z-direction. As such, the units we use are the standard three-dimensional SI units.

Table 1: List of symbols and their corresponding SI units.

Symbol	Description	Units
$\boldsymbol{x}$	Eulerian position	m
$\boldsymbol{X}$	Lagrangian position	m
$\boldsymbol{u}$	Displacement	m
$\boldsymbol{v}$	Displacement velocity	$\mathrm{m\ s^{-1}}$
t	Time	s
$\boldsymbol{g}$	Body force density	$\mathrm{N\ m^{-3}}$
$\boldsymbol{\tau}$	Traction vector	Pa
σ_{ij}	Stress	Pa
$\boldsymbol{n}$	Unit normal vector	1
ε_{ij}	Strain	1
μ_ε, λ_ε	Lamé's first and second parameter	Pa
E	Youngs modulus	Pa
ν	Poisson ratio	1
ρ	Density	$\mathrm{kg\ m^{-2}}$
μ_v, λ_v	Viscosities of viscoelastic medium	Pa s
p	Poro-fluid pressure	Pa
Φ	Fraction of space occupied by solid material	1
k_{perm}	Permeability of porous medium	$\mathrm{m^2}$
μ_{dyn}	Dynamic viscosity of poro-fluid	Pa s
c	Oxygen (nutrient) concentration	$\mathrm{Mol\ m^{-3}}$
η	Ratio of created tissue per existing tissue	1
λ_N	Nutrient diffusion coefficient	$\mathrm{m^2\ s^{-1}}$
N_{G}	Tumor growth rate	$\mathrm{s^{-1}}$
$G_{\max}$	Maximum growth rate	$\mathrm{s^{-1}}$
c_G	Oxygen concentration at which growth rate is half of $G_{\max}$	$\mathrm{Mol\ m^{-3}}$
r_{d}	Tumor death rate	$\mathrm{s^{-1}}$
N_{A}	Nutrient absorption growth rate	$\mathrm{s^{-1}}$
$A_{\max}$	Maximum nutrient absorption rate	$\mathrm{s^{-1}}$
c_A	Oxygen concentration at which absorption rate is half of $A_{\max}$	$\mathrm{Mol\ m^{-3}}$

1 Introduction

The development of a cancerous tumor can be divided into three broad stages [19]. First, there is the initiation or transformation stage. This is a process in which normal cells are altered such that they acquire the potential for autonomous growth and are able to form tumors. Initiation is caused by mutations in the genetic material, DNA. These mutations can occur spontaneously, or they can be induced by environmental agents such as radiation, viruses or chemicals. It has also been shown that people with so-called germline mutations in specific genes leading to hereditary predisposition are more likely to develop cancer. The second stage of tumor development is growth. This stage consists of the two processes, promotion and progression. In promotion, initiated cells can expand and resist mechanisms that would otherwise curtial further development of a tumor. Then, during the progression process, the tumor starts growing rapidly. Tumor growth is dependent on the presence of oxygen and glucose. The final stage of cancer development is called metastasis, which occurs when cancer cells break away from the initial tumor and enter the circulatory system. New tumors can then be formed at different locations in the body.

In this work, we focus on the mechanical aspects of the growth stage of tumor development, specifically the tumor progression process. To this end, a tumor growth model is developed based on the novel combination of morpho- and poroelasticity. Poroelasticity provides us with a mathematical framework that models the interaction of fluid and solid phases in a porous medium. An intuitive way to think out this is to consider a dry sponge that is submerged into a body of water. Due to the water entering the sponge, it will expand. In the field of biomechanics, poroelasticity has been used extensively. For example, to model the flow of blood through arteries [3], bone deformations [5], and of course tumor growth [7, 14]. The addition of morphoelasticity a novel approach. Morphoelasticity is the theory of elastic growth, hence it allows us to take into account permanent deformations of tissue.

In the work done by Eline Kleimann [10] and Daan Smits [16] in their master graduation projects, morphoelasticity is applied to the healing of burn wounds. Their work is based on the PhD thesis of Daniël Koppenol [11], in which a number of different biomedical models are developed to study the healing and treatment of dermal wounds. Although we will apply morphoelasticity to the different field of tumor growth, the underlying mathematics are largely the same. As a tumor grows, it can push the surrounding tissue away. This can lead to complications. For example, a brain tumor exerts force on the surrounding brain tissue, possibly causing permanent damage. Even after a tumor has been surgically removed, the surrounding tissue might be permanently deformed or damaged. The motivation for using a morphoelastic approach is to model these permanent deformations.

In addition to the combination of poro- and morphoelasticity, which we will call the mechanical part of the tumor growth model, we must also consider some biological aspects of tumor growth. For example, a tumor needs nutrients to grow. The biochemical part of the model is based on the work of Tiina Roose et al. [14], in which poroelasticity is coupled with nutrient transport to model the growth of spheroid tumor cells.

This project has a number of goals. First, we want to investigate whether it is possible to construct a finite element model that combines morpo- and poroelasticity. Since this combination is a novel approach, it is unclear whether a standard finite element approach will work. The second goal is to combine this

mechanical model with a biochemical model based on [14].

1.1 Thesis Outline

The outline of this thesis is briefly discussed in the following section. It is divided into two parts. In the first part we develop finite element schemes for existing elastic models. The complexity of these models increases, leading up to a morphoelastic model at the end of Chapter 2, and a visco- poroelastic model at the end of Chapter 3. These models form the building blocks of the eventual tumor growth model, in which morpho- and poroelasticity are combined. The second part of this thesis contains new work. First we show the derivation of the stabilization parameter for a visco- poroelastic system. Next a finite element model for a morpho-poroelastic model is developed. In Chapter 5 morpho-poroelasticity is used as the backbone of the tumor growth model. We give a short summary of the chapters.

- **Part I**
  - **Chapter 2:** A finite element model is developed for increasingly complex elastic models. We start with a regular elasticity equation, then move on to viscoelasticity in which the intertial forces must be considered. Finally, we arrive at morphoelasticity. The model we develop here is effectively the same as the one used in [16]. Each section has a similar setup. First the weak form is derived, then the Galerkin equations are constructed and finally some numerical experiments are run to test the models. In the derivation of the weak forms, a number of theorems are used. We give the proof for some of these theorems, whereas for others we provide a short motivation and cite an external source that contains the proof.
  - **Chapter 3:** The focus is shifted towards poroelasticity. First, a one-dimensional system called the Terzaghi problem is analyzed. It is well known that even for this simple problem the solution can contain non-physical oscillations. We reproduce these oscillations and investigate their severity when a different type of element is used (Taylor-Hood). Next, two-dimensional poroelastic models are developed. Oscillations also occur in these cases. Different methods of stabilization are compared.
- **Part II**
  - **Chapter 4:** We investigate in more detail one of the methods used to stabilize a poroelastic system: diffusive stabilization. In this method a tuning parameter must be chosen, which has a big effect on the performance. In literature such as [1, 13] an optimal value of this parameter is shown for the one-dimensional Terzaghi problem. We prove that this parameter is also (approximately) optimal for a visco- and morpho-poroelastic system. To this end, we use an novel technique which can also be used to find the exact expression of the inverse of a Laplace matrix. This tangential result is explored in Appendix B. In the latter part of this chapter, the viscoelastic and morphoelastic models from Sections 2.3 and 2.4 are combined with poroelasticity. The results on stabilization can also be applied to the morpho-poroelastic model.
  - **Chapter 5:** The morpho-poroelastic model is combined with a biochemical tumor growth model from [14]. First the behaviour of the newly introduced biochemical unknowns is analyzed by considering zero-dimensional problems. We subsequently build the complete finite element scheme for the tumor growth model, and use it to perform numerical experiments.
- **Appendices**
  - **Appendix A:** This chapter contains the element matrices and vectors for any integral that occurs in the Galerkin equations.
  - **Appendix B:** We use the same approach as in Chapter 4 to find an exact expression for the inverse of the Laplace matrix in one dimension. Some additional results are shown, such as the connection to Green's functions and higher order problems.

1.2 A Note on Implementation

The finite element models are implemented in an object oriented fashion using Python 3.6. We make heavy use of the library NumPy, which allows us to work efficiently with vectors. In addition to NumPy, the sparse matrix library from SciPy is used. Since all finite element coefficient matrices are constructed

by looping over all elements, the construction of a sparse matrix object will not be faster than a regular matrix. However, solving the resulting linear system is much faster for certain classes of sparse matrices. We use the CSR (compressed row storage) class. All figures shown in this work are created using the Matplotlib visualization library.

For generating the finite element meshes the library 'dmsh' is used [15]. It supports automatically generated meshes for all kinds of geometries in two dimensions. The meshes it produces are of high quality, in the sense that every triangular element is close to an equilateral triangle. Moreover, dmsh has a user-friendly interface due to its pure Python source code. Unfortunately, mesh generation is quite slow and requires a lot of memory. For these reasons, we do not make use of remeshing. Thus only at the start of a simulation dmsh is used to generate an initial mesh. During the simulation grid nodes are able to move, causing elements to deform. It could be beneficial to create a new mesh with respect to the updated domain.

Part I

2

A Sequence of Models for Elasticity

In this chapter three finite element models are constructed for various elastic systems in two dimensions. We first briefly discuss the basic mathematical framework used to model stress and strain. We subsequently consider the elasticity equation, in which the reaction to a force is instantaneous. A viscous term is then added to expand the model to a viscoelastic system. Finally, we consider the morphoelastic system. Here the strain tensor is made time-dependent and must statisfy a partial differential equation. The right-hand side of this equation is the growth tensor, which allows us to model permanent deformations of the material. Each section has a similar structure. We begin by discussing the mathematical model and present the corresponding equations. Then the weak form is constructed and the corresponding Galerking equations are derived, leading up to a complete algebraic system of equations. We end each section by performing some numerical experiments to test the finite element model. The last section of this chapter contains a comparison between the three models.

2.1 Stress and Strain

We first define the notions of stress and strain, and show the derivation of the force balance equation. Stress is a quantity that measures the internal forces in a material. At a given point in the material, the component σ_{ij} denotes the stress along the i'th coordinate direction acting on a surface of which the normal points in the j'th coordinate direction. Hence, in three dimensions, the stress in a point is completely defined by nine components: there are three normal directions with each three stress components. The nine components are collected in the stress tensor:

$$\boldsymbol{\sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{pmatrix}. \tag{2.1}$$

In two dimensions, there are two normal directions, hence the stress tensor becomes a 2×2 array:

$$\boldsymbol{\sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{pmatrix}. \tag{2.2}$$

The components σ_{ii} are called normal stresses, they result from a force applied perpendicular to a surface. The other components σ_{ij} for $i \neq j$ are called the shear stresses. They result from a force applied parallel to a surface. Given a normalized direction vector $\boldsymbol{n}$, the traction vector $\boldsymbol{\tau}^{(\boldsymbol{n})}$ acting on the surface normal to $\boldsymbol{n}$ can be computed using the stress tensor:

$$\boldsymbol{\tau}^{(\boldsymbol{n})} = \boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{\sigma} \cdot \boldsymbol{n}. \tag{2.3}$$

We will use the two notations $\boldsymbol{n}\cdot\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}\cdot\boldsymbol{n}$ interchangeably, they both denote the standard matrix-vector product of $\boldsymbol{\sigma}$ and $\boldsymbol{n}$. Thus:

$$\tau_i^{(\boldsymbol{n})} = \sum_j \sigma_{ij} n_j. \tag{2.4}$$

Consider a closed volume Ω with piecewise smooth boundary Γ. Let $\boldsymbol{n}$ denote the outward pointing unit normal vector to Γ. Assume that Ω is subject to a body force $\boldsymbol{g}$. By Newton's third law, the action of

the body force will result in an opposite traction force on the boundary:

$$-\int_\Gamma \boldsymbol{n}\cdot\boldsymbol{\sigma}\,\mathrm{d}\Gamma = \int_\Omega \boldsymbol{g}\,\mathrm{d}\Omega. \tag{2.5}$$

We apply Gauss' divergence theorem to obtain

$$-\int_\Omega \nabla\cdot\boldsymbol{\sigma}\,\mathrm{d}\Omega = \int_\Omega \boldsymbol{g}\,\mathrm{d}\Omega. \tag{2.6}$$

If this holds for every volume, then the above integral turns into a pointwise equality:

$$-\nabla\cdot\boldsymbol{\sigma} = \boldsymbol{g}. \tag{2.7}$$

This equation is the momentum balance equation. All models for elasticity are based on this equation. Their differences lie in the choice of the stress tensor $\boldsymbol{\sigma}$. In the purely elastic model, $\boldsymbol{\sigma}$ only contains an elastic component, whereas in the visco- and morphoelastic models, an intertial component involving the displacement velocity is also included. The elastic component of the stress tensor is given in terms of the strain tensor. While there are many different definitions of strain, we will the Eulerian strain, given by:

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\top\right). \tag{2.8}$$

Here $\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X}$ is the displacement vector, it relates the initial position of a particle $\boldsymbol{X}$ to its current position $\boldsymbol{x}$. The gradient ∇ is relative to the current coordinates. In Section 2.3.1 we will give more details regarding the different coordinate systems. It is important to note that $\boldsymbol{\varepsilon}$ is linear with respect to the partial derivatives of $\boldsymbol{u}$.

2.2 Pure Elasticity

We will first consider pure elasticity on a fixed domain Ω. The equations that we consider are stationary, meaning that there will be no timestepping required. The momentum balance equation on a fixed domain Ω is given by:

$$-\nabla\cdot\boldsymbol{\sigma}(\boldsymbol{u}) = \boldsymbol{g}, \tag{2.9}$$

where $\boldsymbol{g}$ is a body force, and the stress tensor is given by:

$$\begin{aligned}\boldsymbol{\sigma}(\boldsymbol{u}) &= \frac{1}{2}\mu_\varepsilon\left(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\top\right) + \lambda_\varepsilon(\nabla\cdot\boldsymbol{u})\boldsymbol{I}\\ &= \mu_\varepsilon\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda_\varepsilon\mathrm{tr}(\boldsymbol{\varepsilon}(\boldsymbol{u}))\boldsymbol{I}\end{aligned} \tag{2.10}$$

Here the strain tensor $\boldsymbol{\varepsilon}$ is defined in (2.8). The constants μ_ε and λ_ε are called the Lamé constants. They relate to the Young's modulus E and Poisson ratio ν in the following way:

$$\mu_\varepsilon = \frac{E}{1+\nu}, \qquad \lambda_\varepsilon = \frac{E\nu}{(1+\nu)(1-2\nu)}. \tag{2.11}$$

We consider a Neumann boundary condition of the form

$$\boldsymbol{\sigma}\cdot\boldsymbol{n} = \boldsymbol{\tau}, \tag{2.12}$$

which represents a shear force on the boundary. A homogeneous Dirichlet boundary conditions of the form

$$\boldsymbol{u} = \boldsymbol{0} \tag{2.13}$$

can also be imposed on part of the boundary. This corresponds to fixing this part of the boundary, such that it can not move as a result of the applied force. To summarize, we will solve the following boundary value problem:

$$-\nabla\cdot\boldsymbol{\sigma}(\boldsymbol{u}) = \boldsymbol{g}, \quad \text{in } \Omega, \tag{2.14}$$

$$\boldsymbol{\sigma}\cdot\boldsymbol{n} = \boldsymbol{\tau}, \quad \text{on } \Gamma_1, \tag{2.15}$$

$$\boldsymbol{u} = \boldsymbol{0}, \quad \text{on } \Gamma_2. \tag{2.16}$$

The boundary Γ of Ω consists of a 'Neumann'-part Γ_1 and a 'Dirichlet'-part Γ_2, such that $\Gamma_1 \cap \Gamma_2 = \varnothing$.

2.2.1 Weak Formulation

We derive the weak form of the above equation. Let $\boldsymbol{H}^1(\Omega)$ be the space of all functions $\boldsymbol{u} : \Omega \to \mathbb{R}^2$ such that $\boldsymbol{u} \in \boldsymbol{L}^2(\Omega)$ and all its partial derivatives are also in $\boldsymbol{L}^2(\Omega)$. Let $\boldsymbol{w}$ be a test function in $\boldsymbol{H}^1(\Omega)$ such that $\boldsymbol{w}|_{\Gamma_2} = \boldsymbol{0}$. Take the inner product of $\boldsymbol{w}$ with the equation and integrate over Ω:

$$-\int_\Omega \boldsymbol{w} \cdot (\nabla \cdot \boldsymbol{\sigma}) \, \mathrm{d}\Omega = \int_\Omega \boldsymbol{w} \cdot \boldsymbol{g} \, \mathrm{d}\Omega. \tag{2.17}$$

The left hand-side can be expanded using Green's formula and Gauss' divergence theorem:

$$\begin{aligned} -\int_\Omega \boldsymbol{w} \cdot (\nabla \cdot \boldsymbol{\sigma}) \, \mathrm{d}\Omega &= -\int_\Omega \nabla \cdot (\boldsymbol{w} \cdot \boldsymbol{\sigma}) - \nabla \boldsymbol{w} : \boldsymbol{\sigma} \, \mathrm{d}\Omega \\ &= \int_\Omega \nabla \boldsymbol{w} : \boldsymbol{\sigma} \, \mathrm{d}\Omega - \int_\Gamma (\boldsymbol{w} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{n} \, \mathrm{d}\Gamma \\ &= \int_\Omega \nabla \boldsymbol{w} : \boldsymbol{\sigma} \, \mathrm{d}\Omega - \int_{\Gamma_1} \boldsymbol{w} \cdot \boldsymbol{\tau} \, \mathrm{d}\Gamma \end{aligned} \tag{2.18}$$

In the last step, we used the fact that $\boldsymbol{w}|_{\Gamma_2} = \boldsymbol{0}$ and that $(\boldsymbol{w} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{n} = \boldsymbol{w} \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{n}) = \boldsymbol{w} \cdot \boldsymbol{\tau}$ on Γ_1. We obtain the following weak formulation:

Find $\boldsymbol{u} \in \boldsymbol{H}^1_{\Gamma_2}(\Omega)$ such that for all $\boldsymbol{w} \in \boldsymbol{H}^1_{\Gamma_2}(\Omega)$ we have:

$$\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{w}) : \boldsymbol{\sigma}(\boldsymbol{u}) \, \mathrm{d}\Omega = \int_\Omega \boldsymbol{w} \cdot \boldsymbol{g} \, \mathrm{d}\Omega + \int_{\Gamma_1} \boldsymbol{w} \cdot \boldsymbol{\tau} \, \mathrm{d}\Gamma. \tag{2.19}$$

Here $\boldsymbol{H}^1_{\Gamma_2}(\Omega)$ is the linear subspace of $\boldsymbol{H}^1(\Omega)$ containing functions that vanish on Γ_2.

We write $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u})$ to stress[1] that $\boldsymbol{\sigma}$ is a functional acting on $\boldsymbol{u}$. Also note that in the left-hand side, $\nabla \boldsymbol{w}$ has turned into $\boldsymbol{\varepsilon}(\boldsymbol{w})$. Due to the stress tensor $\boldsymbol{\sigma}$ being symmetric, the double dot products are equal:

$$\begin{aligned} \boldsymbol{\varepsilon}(\boldsymbol{w}) : \boldsymbol{\sigma}(\boldsymbol{u}) &= \frac{1}{2} \left(\nabla \boldsymbol{w} + \nabla \boldsymbol{w}^\top \right) : \boldsymbol{\sigma}(\boldsymbol{u}) \\ &= \frac{1}{2} \nabla \boldsymbol{w} : \boldsymbol{\sigma}(\boldsymbol{u}) + \frac{1}{2} \nabla \boldsymbol{w}^\top : \boldsymbol{\sigma}(\boldsymbol{u}) \\ &= \frac{1}{2} \nabla \boldsymbol{w} : \boldsymbol{\sigma}(\boldsymbol{u}) + \frac{1}{2} \nabla \boldsymbol{w} : \boldsymbol{\sigma}(\boldsymbol{u})^\top \\ &= \nabla \boldsymbol{w} : \boldsymbol{\sigma}(\boldsymbol{u}) \end{aligned} \tag{2.20}$$

The element matrices and vectors corresponding to the integrals in (2.19) are given in Section A.1.

2.2.2 Galerkin Equations

We derive the Galerkin equations corresponding to the weak form (2.19), which is equivalent to

$$\int_\Omega \nabla \boldsymbol{w} : \boldsymbol{\sigma}(\boldsymbol{u}) \, \mathrm{d}\Omega = \int_\Omega \boldsymbol{w} \cdot \boldsymbol{g} \, \mathrm{d}\Omega + \int_{\Gamma_1} \boldsymbol{w} \cdot \boldsymbol{\tau} \, \mathrm{d}\Gamma. \tag{2.21}$$

From a computational perspective, it is easier to use $\nabla \boldsymbol{w}$ instead of $\boldsymbol{\varepsilon}(\boldsymbol{w})$. Let $\boldsymbol{u}_h$ be the finite element solution, and let φ_j be a scalar-valued basis function in mesh point j. We write

$$\boldsymbol{u}_h(\boldsymbol{x}) = \sum_{j=1}^{n} \left(u_j^1 \boldsymbol{\varphi}_j^1(\boldsymbol{x}) + u_j^2 \boldsymbol{\varphi}_j^2(\boldsymbol{x}) \right), \tag{2.22}$$

where the indices $j = 1, \ldots, n$ correspond to grid nodes that are not on the fixed part of the boundary Γ_2. The vector valued basis functions $\boldsymbol{\varphi}_j^1$ and $\boldsymbol{\varphi}_j^2$ are given by:

$$\boldsymbol{\varphi}_j^1 = \begin{pmatrix} \varphi_j \\ 0 \end{pmatrix}, \quad \text{and} \quad \boldsymbol{\varphi}_j^2 = \begin{pmatrix} 0 \\ \varphi_j \end{pmatrix}. \tag{2.23}$$

[1] Pun not intended.

Substitute $\boldsymbol{u}_h$ into (2.21) to obtain

$$\sum_{j=1}^{n}\left[u_j^1\int_\Omega \nabla\boldsymbol{w}:\boldsymbol{\sigma}(\boldsymbol{\varphi}_j^1)\,\mathrm{d}\Omega + u_j^2\int_\Omega \nabla\boldsymbol{w}:\boldsymbol{\sigma}(\boldsymbol{\varphi}_j^2)\,\mathrm{d}\Omega\right] = \int_\Omega \boldsymbol{w}\cdot\boldsymbol{g}\,\mathrm{d}\Omega + \int_{\Gamma_1}\boldsymbol{w}\cdot\boldsymbol{\tau}\,\mathrm{d}\Gamma. \tag{2.24}$$

Setting $\boldsymbol{w}$ equal to $\boldsymbol{\varphi}_i^1$ and $\boldsymbol{\varphi}_i^2$ for $i=1,\ldots,n$ yields $2n$ linear equations for the coefficients u_j^1 and u_j^2. These equations are of the form

$$\left\{\begin{array}{rcl} S^{11}\boldsymbol{u}^1 + S^{12}\boldsymbol{u}^2 &=& \boldsymbol{g}^1+\boldsymbol{\tau}^1 \\ S^{21}\boldsymbol{u}^1 + S^{22}\boldsymbol{u}^2 &=& \boldsymbol{g}^2+\boldsymbol{\tau}^2 \end{array}\right., \quad \text{or} \quad \begin{pmatrix} S^{11} & S^{12}\\ S^{21} & S^{22}\end{pmatrix}\begin{pmatrix}\boldsymbol{u}^1\\ \boldsymbol{u}^2\end{pmatrix} = \begin{pmatrix}\boldsymbol{g}^1+\boldsymbol{\tau}^1\\ \boldsymbol{g}^2+\boldsymbol{\tau}^2\end{pmatrix}, \tag{2.25}$$

where the latter representation is in block-matrix form. The coefficient vectors $\boldsymbol{u}^1$ and $\boldsymbol{u}^2$ contain the unknowns:

$$\boldsymbol{u}^1 = \left(u_1^1, u_2^1, \ldots, u_n^1\right)^\top, \quad \boldsymbol{u}^2 = \left(u_1^2, u_2^2, \ldots, u_n^2\right)^\top. \tag{2.26}$$

Moreover, the vectors $\boldsymbol{g}^1$ and $\boldsymbol{g}^2$ contain the integrals involving the body force $\boldsymbol{g}$, whereas the vectors $\boldsymbol{\tau}^1$ and $\boldsymbol{\tau}^2$ contain integrals involving the shear force $\boldsymbol{\tau}$. The system can be written in the even shorter notation $S\boldsymbol{u} = \boldsymbol{g}+\boldsymbol{\tau}$, where $\boldsymbol{u} = (\boldsymbol{u}^1, \boldsymbol{u}^2)^\top$, etc. To compute the entries of the blocks S^{11}, S^{12}, S^{21} and S^{22}, we must first compute the gradients of the $\boldsymbol{\varphi}_j^1$ and $\boldsymbol{\varphi}_j^2$:

$$\nabla\boldsymbol{\varphi}_j^1 = \begin{pmatrix}\dfrac{\partial\varphi_j}{\partial x} & \dfrac{\partial\varphi_j}{\partial y}\\ 0 & 0\end{pmatrix}, \qquad \nabla\boldsymbol{\varphi}_j^2 = \begin{pmatrix}0 & 0\\ \dfrac{\partial\varphi_j}{\partial x} & \dfrac{\partial\varphi_j}{\partial y}\end{pmatrix}. \tag{2.27}$$

From which it follows that

$$\mathrm{sym}(\nabla\boldsymbol{\varphi}_j^1) = \begin{pmatrix}\dfrac{\partial\varphi_j}{\partial x} & \dfrac{1}{2}\dfrac{\partial\varphi_j}{\partial y}\\ \dfrac{1}{2}\dfrac{\partial\varphi_j}{\partial y} & 0\end{pmatrix}, \qquad \mathrm{sym}(\nabla\boldsymbol{\varphi}_j^2) = \begin{pmatrix}0 & \dfrac{1}{2}\dfrac{\partial\varphi_j}{\partial x}\\ \dfrac{1}{2}\dfrac{\partial\varphi_j}{\partial x} & \dfrac{\partial\varphi_j}{\partial y}\end{pmatrix}. \tag{2.28}$$

Also note:

$$\nabla\cdot\boldsymbol{\varphi}_j^1 = \frac{\partial\varphi_j}{\partial x}, \qquad \nabla\cdot\boldsymbol{\varphi}_j^2 = \frac{\partial\varphi_j}{\partial y}. \tag{2.29}$$

The elements of the S-blocks are then given by:

$$S_{ij}^{11} = \int_\Omega \nabla\boldsymbol{\varphi}_i^1:\boldsymbol{\sigma}(\boldsymbol{\varphi}_j^1)\,\mathrm{d}\Omega = \int_\Omega (\mu_\varepsilon+\lambda_\varepsilon)\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y}\,\mathrm{d}\Omega, \tag{2.30}$$

$$S_{ij}^{12} = \int_\Omega \nabla\boldsymbol{\varphi}_i^1:\boldsymbol{\sigma}(\boldsymbol{\varphi}_j^2)\,\mathrm{d}\Omega = \int_\Omega \lambda_\varepsilon\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x}\,\mathrm{d}\Omega, \tag{2.31}$$

$$S_{ij}^{21} = \int_\Omega \nabla\boldsymbol{\varphi}_i^2:\boldsymbol{\sigma}(\boldsymbol{\varphi}_j^1)\,\mathrm{d}\Omega = \int_\Omega \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \lambda_\varepsilon\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x}\,\mathrm{d}\Omega, \tag{2.32}$$

$$S_{ij}^{22} = \int_\Omega \nabla\boldsymbol{\varphi}_i^2:\boldsymbol{\sigma}(\boldsymbol{\varphi}_j^2)\,\mathrm{d}\Omega = \int_\Omega \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + (\mu_\varepsilon+\lambda_\varepsilon)\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y}\,\mathrm{d}\Omega. \tag{2.33}$$

The elements of the right-hand side vectors are given by:

$$g_i^1 = \int_\Omega \varphi_i g^1\,\mathrm{d}\Omega, \quad g_i^2 = \int_\Omega \varphi_i g^2\,\mathrm{d}\Omega \tag{2.34}$$

$$\tau_i^1 = \int_{\Gamma_1}\varphi_i\tau^1\,\mathrm{d}\Gamma, \quad \tau_i^2 = \int_{\Gamma_1}\varphi_i\cdot\tau^2\,\mathrm{d}\Gamma \tag{2.35}$$

2.2.3 Numerical Experiments

Let us test the finite element implementation on a simple situation. We consider a cross section of a beam of length L and height 1 that is fixed to a rigid wall at its left boundary. We assume the beam has uniform density and is subject to gravity. We model gravity using the body force

$$\boldsymbol{g}(x,y) = \begin{pmatrix}0\\ -F\end{pmatrix},$$

for some constant F. At the left boundary ($x = 0$), the homogeneous Dirichlet boundary condition $\boldsymbol{u} = \mathbf{0}$ is imposed. At all other boundaries, we assume there is zero stress in the normal direction, yielding the boundary condition $\boldsymbol{\sigma} \cdot \boldsymbol{n} = \mathbf{0}$. Having fully specified the problem, we apply the finite element model constructed in the previous section. For generating the following figures, we have fixed $\mu_e = 0.5$, $\lambda_e = 0$ and vary L and F. Figures 2.1 and 2.2 show the solution for $L = 2$ and $F = 0.01$. A mesh size of $h_{\max} = 0.1$ has been used.

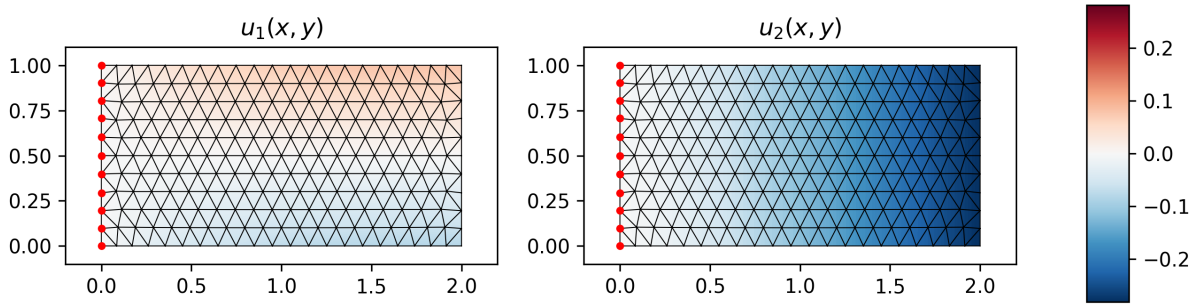


Figure 2.1: Contour plots of both components of the finite element solution for $L = 2$ and $F = 0.01$.

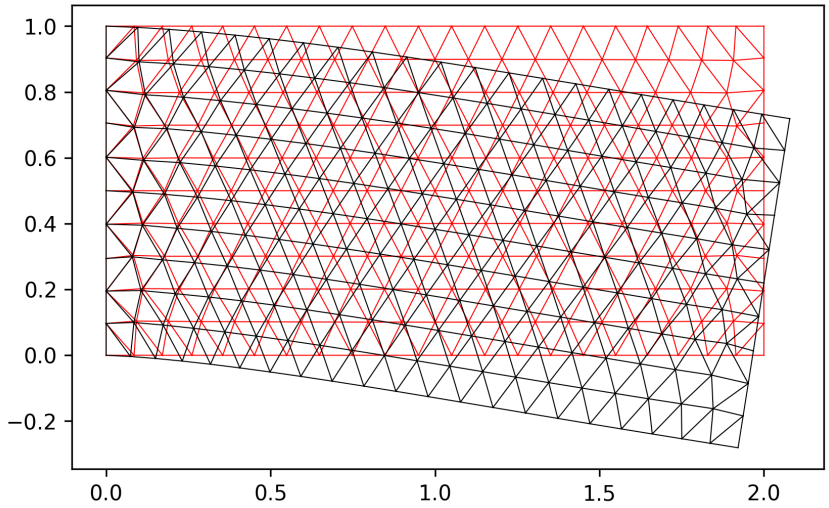


Figure 2.2: Mesh displacement corresponding to the solution in Figure 2.1.

We now double the magnitude of the body force, while keeping $L = 2$ fixed.

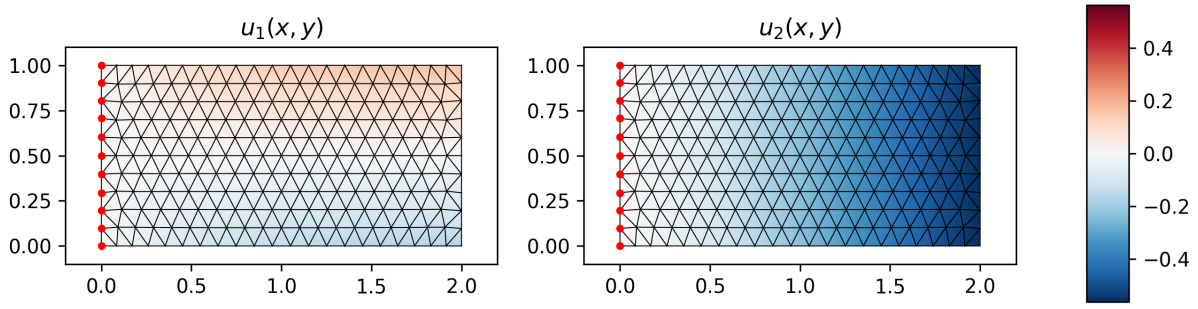


Figure 2.3: Contour plots of both components of the finite element solution for $L = 2$ and $F = 0.02$.

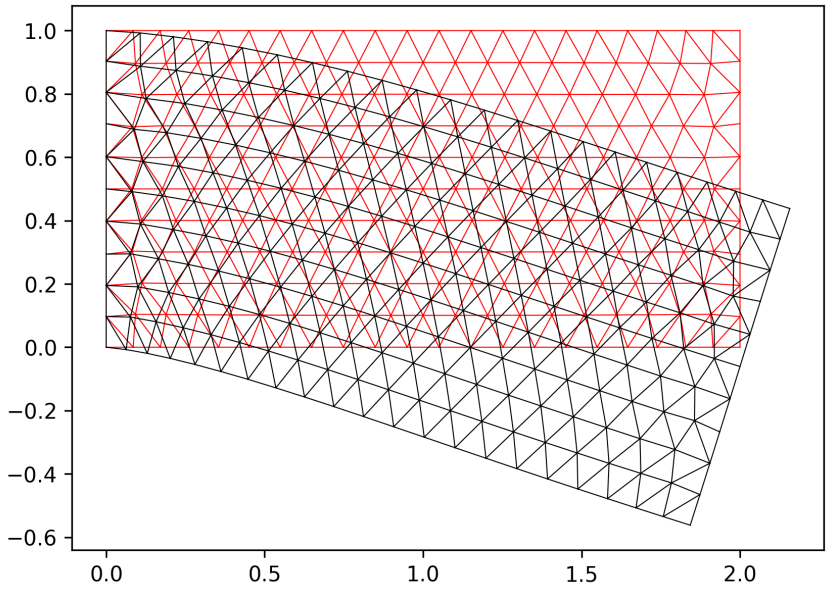


Figure 2.4: Mesh displacement corresponding to the solution in Figure 2.3.

2.3 Viscoelasticity

We add a inertial- or viscous term to the stationary elasticity equation, which will make the equations time-dependent. The viscoelastic momentum-balance equation is given by

$$\rho\left(\frac{D\boldsymbol{v}}{Dt} + (\nabla\cdot\boldsymbol{v})\boldsymbol{v}\right) - \nabla\cdot\boldsymbol{\sigma}(\boldsymbol{u},\boldsymbol{v}) = \boldsymbol{g}, \tag{2.36}$$

where $\boldsymbol{v}$ is the displacement velocity as defined in (2.44). Furthermore, ρ is the material density, and $\boldsymbol{g}$ is again a body force. Due to the inclusion of intertial forces, the stress tensor must also be adapted. It now consists of a purely elastic part and a viscous part:

$$\boldsymbol{\sigma}(\boldsymbol{u},\boldsymbol{v}) = \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{u}) + \boldsymbol{\sigma}_{\text{vis}}(\boldsymbol{v}), \tag{2.37}$$

where

$$\boldsymbol{\sigma}_{\text{el}}(\boldsymbol{u}) = \frac{1}{2}\mu_\varepsilon\left(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^\top\right) + \lambda_\varepsilon(\nabla\cdot\boldsymbol{u})\boldsymbol{I} = \mu_\varepsilon\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda_\varepsilon\text{tr}(\boldsymbol{\varepsilon}(\boldsymbol{u}))\boldsymbol{I}, \tag{2.38}$$

$$\boldsymbol{\sigma}_{\text{vis}}(\boldsymbol{v}) = \frac{1}{2}\mu_v\left(\nabla\boldsymbol{v} + \nabla\boldsymbol{v}^\top\right) + \lambda_v(\nabla\cdot\boldsymbol{v})\boldsymbol{I}. \tag{2.39}$$

Equation (2.36) is solved on a moving domain $\Omega(t)$, it is defined as:

$$\Omega(t) = \{\boldsymbol{X} + \boldsymbol{u}(\boldsymbol{x}(\boldsymbol{X},t),t) \mid \boldsymbol{X} \in \Omega(0)\}. \tag{2.40}$$

Furthermore, the gradient ∇ is relative to the current coordinates. In developing the finite element model we therefore have two choices. As a first option we can use a fixed mesh. In this setting, to compute the gradient with respect to the deformed domain, we must use a mapping that relates current coordinates to the initial coordinates in the fixed mesh. This is called a Eulerian approach. The second option would be to make the mesh dynamic: the grid nodes follow the flow of the domain. In this case we do not need a coordinate mapping, but we need to update the mesh in each timestep. The second method is called the Lagrangian approach, which is the approach that we choose to use. In Section 2.3.1 we give a brief overview of the mathematics behind the distinction between Eulerian and Lagrangian approaches.

2.3.1 Eulerian Versus Lagrangian Coordinates

The finite element model developed in this work is based on the Lagrangian framework. In this section, we will briefly discuss the difference between a Lagrangian and Eulerian specification. We can define the flow of a particle by considering its initial position $\boldsymbol{X}$ and its position at time t, which we denote by $\boldsymbol{x}(\boldsymbol{X},t)$. Note that we can also turn it around; assuming that a particle has position $\boldsymbol{x}$ at some time t,

we can find its initial position $\boldsymbol{X}(\boldsymbol{x}, t)$.

Let φ be a physical quantity that depends on space and time. In a Eulerian framework we consider the evolution of φ in a fixed point and write $\varphi = \varphi(\boldsymbol{x}, t)$. In the Lagrangian framework, the value of φ is considered along the trajectory of a particle. Thus φ effectively becomes a function of the initial position of the particle that is being followed: $\widetilde{\varphi}(\boldsymbol{X}, t) = \varphi(\boldsymbol{x}(\boldsymbol{X}, t), t)$. We use $\widetilde{\varphi}$ instead of φ to emphasize it being a different function in the mathematical sense. Care must be taken in evaluating the rate of change of φ. The standard partial time-derivative $\partial\varphi/\partial t$ measures the rate of change of φ at a fixed position. The full time-derivative takes into account the movement of particles:

$$\frac{\mathrm{d}\varphi}{\mathrm{d}t}(\boldsymbol{x}, t) = \frac{\partial\varphi}{\partial t} + \frac{\partial\varphi}{\partial\boldsymbol{x}} \cdot \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t}. \tag{2.41}$$

The above expression is called the *material derivative*. By introducing the notation $\mathrm{d}\boldsymbol{x}/\mathrm{d}t = \boldsymbol{v}$ and writing $\partial\varphi/\partial\boldsymbol{x} = \nabla_{\boldsymbol{x}}\varphi$, the material derivative is denoted by

$$\frac{D\varphi}{Dt} = \frac{\partial\varphi}{\partial t} + \boldsymbol{v} \cdot \nabla_{\boldsymbol{x}}\varphi. \tag{2.42}$$

We write $\nabla_{\boldsymbol{x}}$ to emphasize we are dealing with the Eulerian gradient: the spatial derivative are taken with respect to the current particle locations $\boldsymbol{x}$. In the rest of this work, the subscript is omitted and we simply use ∇ to denote the Eulerian gradient. The Eulerian and Lagriangian coordinates are related through the displacement vector $\boldsymbol{u}$, which is defined as the difference between the current and initial position of a particle. Thus, in the Lagriangian framework we have:

$$\boldsymbol{u}(\boldsymbol{X}, t) = \boldsymbol{x}(\boldsymbol{X}, t) - \boldsymbol{X}. \tag{2.43}$$

From this definition it follows that

$$\frac{D\boldsymbol{u}}{Dt} = \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{v}, \tag{2.44}$$

since $\mathrm{d}\boldsymbol{X}/\mathrm{d}t = \boldsymbol{0}$. Therefore, $\boldsymbol{v}$ is called the *displacement velocity*. The relation between the initial position $\boldsymbol{X}$ and the current position $\boldsymbol{x}$ is encoded in the *deformation tensor*, which is given by

$$\boldsymbol{F} = \frac{\partial\boldsymbol{x}}{\partial\boldsymbol{X}}. \tag{2.45}$$

If $\boldsymbol{F}$ is close to the identity matrix, then its inverse is too. In this case, the deformations are small, and we have

$$\begin{aligned}
\frac{D\boldsymbol{u}}{Dt} &= \frac{\partial\boldsymbol{u}}{\partial t} + \nabla\boldsymbol{u} \cdot \boldsymbol{v} \\
&= \frac{\partial\boldsymbol{u}}{\partial t} + \nabla\left(\boldsymbol{x} - \boldsymbol{X}\right) \cdot \boldsymbol{v} \\
&= \frac{\partial\boldsymbol{u}}{\partial t} + \left(\boldsymbol{I} - \boldsymbol{F}^{-1}\right) \cdot \boldsymbol{v} \\
&\approx \frac{\partial\boldsymbol{u}}{\partial t}
\end{aligned} \tag{2.46}$$

Thus, the material derivative of $\boldsymbol{u}$ is closely approximated by the partial time-derivative. In this situation we are dealing with *infinitessimal deformations*. In most of this work, finite deformations are considered, meaning that $\|\boldsymbol{F} - \boldsymbol{I}\| > 0$. Only in the first two sections of Chapter 3 do we consider infinitessimal deformations, which is done to investigate the behaviour of the pressure.

As stated, our finite element model will be developed using a Lagrangian framework. As a result, the mesh nodes of the finite element mesh will follow the displacement velocity field.

2.3.2 Complete Viscoelastic Model Equations

To fully specify the viscoelatic system we need to consider the boundary conditions. We denote the boundary of $\Omega(t)$ by $\Gamma(t)$. The boundary is again split into two disjoint parts: $\Gamma(t) = \Gamma_1(t) \cup \Gamma_2(t)$ such that $\Gamma_1(t) \cap \Gamma_2(t) = \varnothing$ for all $t \geqslant 0$. Since the domain is be moving, it is not straight-forward to determine where one part begins and the other ends. To solve this problem, $\Gamma_1(t)$ is fixed in place by imposed the

boundary condition $\boldsymbol{u} = \boldsymbol{0}$. The other part $\Gamma_2(t)$ is free to move, and may be subject to a shear force τ. In summary, we obtain the following viscoelastic system:

$$\rho\left(\frac{D\boldsymbol{v}}{Dt} + (\nabla \cdot \boldsymbol{v})\boldsymbol{v}\right) - \nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{g}, \qquad \text{in } \Omega(t), \tag{2.47}$$

$$\frac{D\boldsymbol{u}}{Dt} = \boldsymbol{v}, \qquad \text{in } \Omega(t), \tag{2.48}$$

with boundary conditions

$$\boldsymbol{u} = \boldsymbol{0}, \qquad \text{on } \Gamma_1(t), \tag{2.49}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\tau}, \qquad \text{on } \Gamma_2(t). \tag{2.50}$$

In subsequent sections, we derive the weak form and Galerkin equations corresponding to the system. To achieve this, an important result summarized in Theorem 3 is required. This theorem is proved using a number of well-known results, such as Reynold's Transport Theorem and the product rule for the material derivative. Theorem 3 is used numerous times throughout this work, as it allows us to greatly simplify convert complex integrals involving material derivatives.

2.3.3 Weak Formulation

We derive the weak form corresponding to equation (2.47). Let $\boldsymbol{\varphi}$ be a basis function. Take the inner product of $\boldsymbol{\varphi}$ with the momentum-balance equation and integrate over $\Omega(t)$ to obtain the following weak form:

$$\rho\left[\int_{\Omega(t)} \frac{D\boldsymbol{v}}{Dt} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega + \int_{\Omega(t)} (\nabla \cdot \boldsymbol{v})\boldsymbol{v} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega\right] - \int_{\Omega(t)} (\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega = \int_{\Omega(t)} \boldsymbol{f} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega. \tag{2.51}$$

We shall use the following theorems and lemma to manipulate the weak form. First, we state Reynolds Transport Theorem:

Theorem 1 (Reynold's Transport Theorem)**.** *Let $f(\boldsymbol{x}, t)$ be a suffiently smooth scalar-valued function. Let $\Omega(t)$ be a time-dependent domain with piecewise smooth boundary $\Gamma(t)$, and $\boldsymbol{v}(\boldsymbol{x}, t)$ the velocity of a point $\boldsymbol{x} \in \Omega(t)$, then:*

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)} f(\boldsymbol{x}, t) \, \mathrm{d}\Omega = \int_{\Omega(t)} \frac{\partial f}{\partial t}(\boldsymbol{x}, t) \, \mathrm{d}\Omega + \int_{\Gamma(t)} f(\boldsymbol{x}, t) \, \boldsymbol{v} \cdot \boldsymbol{n} \, \mathrm{d}\Gamma,$$

where $\boldsymbol{n}$ is the outward-pointing unit normal vector to $\Gamma(t)$.

We will not give the proof of this well-known theorem. In addition, we need the following result:

Theorem 2 (Dziuk & Elliott)**.** *If φ is a Lagragian basis function, then*

$$\frac{D\varphi}{Dt} = 0.$$

The proof can be found in [6]. A Lagrangian basis function is a Lagrangian interpolation polynomial on an element that interpolates between the vertices. We will demonstrate the validity of the theorem by considering a linear interpolatory function in one dimension. Let $x_1(t)$, $x_2(t)$ be two moving grid nodes. We define

$$\varphi(x; t) = \frac{x_2(t) - x}{x_2(t) - x_1(t)}\varphi_1 + \frac{x - x_1(t)}{x_2(t) - x_1(t)}\varphi_2, \tag{2.52}$$

where φ_1 and φ_2 are constant in time. Thus, φ linearly interpolates the values φ_1 and φ_2 on the moving interval $[x_1(t), x_2(t)]$. It follows that:

$$\frac{\partial \varphi}{\partial x}(x; t) = \frac{\varphi_2 - \varphi_1}{x_2(t) - x_1(t)}, \tag{2.53}$$

and

$$\frac{\partial \varphi}{\partial t}(x; t) = -\frac{\varphi_2 - \varphi_1}{x_2(t) - x_1(t)}\left[\frac{x_2(t) - x}{x_2(t) - x_1(t)}\frac{\mathrm{d}x_1}{\mathrm{d}t}(t) + \frac{x - x_1(t)}{x_2(t) - x_1(t)}\frac{\mathrm{d}x_2}{\mathrm{d}t}(t)\right]. \tag{2.54}$$

We can find the velocity at the point x by considering the velocities of the endpoints:

$$v(x, t) = \frac{x_2(t) - x}{x_2(t) - x_1(t)}\frac{\mathrm{d}x_1}{\mathrm{d}t}(t) + \frac{x - x_1(t)}{x_2(t) - x_1(t)}\frac{\mathrm{d}x_2}{\mathrm{d}t}(t). \tag{2.55}$$

From which it follows that

$$\frac{\partial \varphi}{\partial t}(x;t) = -\frac{\varphi_2 - \varphi_1}{x_2(t) - x_1(t)} v(x,t). \tag{2.56}$$

By the definition of the material derivative and equations (2.53) and (2.56), we get

$$\begin{aligned}\frac{D\varphi}{Dt} &= \frac{\partial \varphi}{\partial t} + v(x,t)\frac{\partial \varphi}{\partial x} \\ &= -\frac{\varphi_2 - \varphi_1}{x_2(t) - x_1(t)} v(x,t) + \frac{\varphi_2 - \varphi_1}{x_2(t) - x_1(t)} v(x,t) = 0,\end{aligned} \tag{2.57}$$

Showing that Theorem 2 indeed holds in this case. It follows immediately from Theorem 2 that for a vector valued Lagrangian basis function $\boldsymbol{\varphi}$ we also have $D\boldsymbol{\varphi}/Dt = \mathbf{0}$. Indeed, we can write $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_d)^\top$, thus

$$\frac{D\boldsymbol{\varphi}}{Dt} = \begin{pmatrix} \frac{D\varphi_1}{Dt} \\ \vdots \\ \frac{D\varphi_d}{Dt} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{2.58}$$

The last ingredient for the proof of Theorem 3 is the product rule for the material derivative.

Lemma 1 (Product Rule)**.** *Let $\Omega(t) \subset \mathbb{R}^d$ be a domain subject to a velocity $\boldsymbol{v}(\boldsymbol{x},t)$. Let $f(\boldsymbol{x},t)$ and $g(\boldsymbol{x},t)$ be two differentiable functions on $\Omega(t)$. Then we have:*

$$\frac{D}{Dt}(\boldsymbol{f} \cdot \boldsymbol{g}) = \frac{D\boldsymbol{f}}{Dt} \cdot \boldsymbol{g} + \boldsymbol{f} \cdot \frac{D\boldsymbol{g}}{Dt}.$$

Proof. The statement is proved using the definition of the material derivative:

$$\begin{aligned}\frac{D}{Dt}(\boldsymbol{f} \cdot \boldsymbol{g}) &= \frac{\partial}{\partial t}(\boldsymbol{f} \cdot \boldsymbol{g}) + \boldsymbol{v} \cdot \nabla(\boldsymbol{f} \cdot \boldsymbol{g}) \\ &= \frac{\partial}{\partial t}\left(\sum_{i=1}^{d} f_i g_i\right) + \boldsymbol{v} \cdot \left(\sum_{i=1}^{d} \nabla(f_i g_i)\right) \\ &= \sum_{i=1}^{d}\left(\frac{\partial f_i}{\partial t} g_i + f_i \frac{\partial g_i}{\partial t}\right) + \sum_{k=1}^{d} v_k \left(\sum_{i=1}^{d} \frac{\partial}{\partial x_k}(f_i g_i)\right) \\ &= \sum_{i=1}^{d}\left(\frac{\partial f_i}{\partial t} g_i + f_i \frac{\partial g_i}{\partial t}\right) + \sum_{i=1}^{d}\sum_{k=1}^{d} v_k \left(\frac{\partial f_i}{\partial x_k} g_i + f_i \frac{\partial g_i}{\partial x_k}\right) \\ &= \sum_{i=1}^{d}\left(\frac{\partial f_i}{\partial t} g_i + f_i \frac{\partial g_i}{\partial t}\right) + \sum_{i=1}^{d}\left(g_i(\boldsymbol{v} \cdot \nabla f_i) + f_i(\boldsymbol{v} \cdot \nabla g_i)\right) \\ &= \sum_{i=1}^{d}\left(g_i\left(\frac{\partial f_i}{\partial t} + \boldsymbol{v} \cdot \nabla f_i\right) + f_i\left(\frac{\partial g_i}{\partial t} + \boldsymbol{v} \cdot \nabla g_i\right)\right) \\ &= \frac{D\boldsymbol{f}}{Dt} \cdot \boldsymbol{g} + \boldsymbol{f} \cdot \frac{D\boldsymbol{g}}{Dt}.\end{aligned}$$

□

Theorems 1, 2 and Lemma 1 are used to prove the following theorem. It is particularly useful for reducing complicated expressions involving the material derivative to easy to discretize terms using Finite Elements.

Theorem 3. *Let $\boldsymbol{f}(\boldsymbol{x},t)$ be a sufficiently smooth vector function. Furthermore, let $\boldsymbol{\varphi}$ be a Lagrangian basis function, and let $\boldsymbol{v}(\boldsymbol{x},t)$ be the velocity of the point $\boldsymbol{x} \in \Omega(t)$. Then:*

$$\int_{\Omega(t)} \left(\frac{D\boldsymbol{f}}{Dt} + (\nabla \cdot \boldsymbol{v})\boldsymbol{f}\right) \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega = \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \boldsymbol{f} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega.$$

Proof. Using Lemma 1 and Theorem 2, we find:

$$\int_{\Omega(t)} \frac{D\boldsymbol{f}}{Dt} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega = \int_{\Omega(t)} \frac{D}{Dt}(\boldsymbol{f} \cdot \boldsymbol{\varphi}) - \boldsymbol{f} \cdot \frac{D\boldsymbol{\varphi}}{Dt} \, \mathrm{d}\Omega = \int_{\Omega(t)} \frac{D}{Dt}(\boldsymbol{f} \cdot \boldsymbol{\varphi}) \, \mathrm{d}\Omega$$

Define $\xi = \boldsymbol{f} \cdot \boldsymbol{\varphi}$, then:

$$\begin{aligned}\int_{\Omega(t)} \frac{D}{Dt}(\boldsymbol{f} \cdot \boldsymbol{\varphi}) \,\mathrm{d}\Omega &= \int_{\Omega(t)} \frac{D\xi}{Dt} \,\mathrm{d}\Omega \\ &= \int_{\Omega(t)} \frac{\partial \xi}{\partial t} + \boldsymbol{v} \cdot \nabla \xi \,\mathrm{d}\Omega \\ &= \int_{\Omega(t)} \frac{\partial \xi}{\partial t} + \nabla \cdot (\xi \boldsymbol{v}) - (\nabla \cdot \boldsymbol{v})\xi \,\mathrm{d}\Omega \\ &= \int_{\Omega(t)} \frac{\partial \xi}{\partial t} - (\nabla \cdot \boldsymbol{v})\xi \,\mathrm{d}\Omega + \int_{\Gamma(t)} \xi(\boldsymbol{v} \cdot \boldsymbol{n}) \,\mathrm{d}\Gamma\end{aligned}$$

Using Theorem 1 we find:

$$\begin{aligned}\int_{\Omega(t)} \left(\frac{D\boldsymbol{f}}{Dt} + (\nabla \cdot \boldsymbol{v})\boldsymbol{f}\right) \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega &= \int_{\Omega(t)} \frac{D\xi}{Dt} + (\nabla \cdot \boldsymbol{v})\xi \,\mathrm{d}\Omega \\ &= \int_{\Omega(t)} \frac{\partial \xi}{\partial t} - (\nabla \cdot \boldsymbol{v})\xi + (\nabla \cdot \boldsymbol{v})\xi \,\mathrm{d}\Omega + \int_{\Gamma(t)} \xi(\boldsymbol{v} \cdot \boldsymbol{n}) \,\mathrm{d}\Gamma \\ &= \int_{\Omega(t)} \frac{\partial \xi}{\partial t} \,\mathrm{d}\Omega + \int_{\Gamma(t)} \xi(\boldsymbol{v} \cdot \boldsymbol{n}) \,\mathrm{d}\Gamma \\ &= \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \xi \,\mathrm{d}\Omega = \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \boldsymbol{f} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega\end{aligned}$$

□

Thus, if we assume that $\boldsymbol{\varphi}$ is a Lagrangian basis function, then by Theorem 3 the weak form (2.51) is equivalent to:

$$\rho \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \boldsymbol{v} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega - \int_{\Omega(t)} (\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega = \int_{\Omega(t)} \boldsymbol{g} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega. \tag{2.59}$$

Using the same derivation as in (2.18), the weak form can be further expanded into

$$\rho \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \boldsymbol{v} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega + \int_{\Omega(t)} \boldsymbol{\sigma}(\boldsymbol{u}, \boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{\varphi}) \,\mathrm{d}\Omega = \int_{\Omega(t)} \boldsymbol{g} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega + \int_{\Gamma_2(t)} \boldsymbol{\tau} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Gamma. \tag{2.60}$$

The stress tensor is given by (2.37). This splitting is useful, because the two tensors $\boldsymbol{\sigma}_{\text{el}}$ and $\boldsymbol{\sigma}_{\text{vis}}$ are very similar to the stress tensor in (2.10). Thus, the element matrix entries are also very similar to those derived in section 2.2.2.

2.3.4 Galerkin Equations

Let $\boldsymbol{u}_h$ and $\boldsymbol{v}_h$ be the Finite Element solutions, we write

$$\boldsymbol{u}_h(\boldsymbol{x}, t) = \sum_{j=1}^{n} u_j^1(t)\boldsymbol{\varphi}_j^1(\boldsymbol{x}; t) + u_j^2(t)\boldsymbol{\varphi}_j^2(\boldsymbol{x}; t), \tag{2.61}$$

$$\boldsymbol{v}_h(\boldsymbol{x}, t) = \sum_{j=1}^{n} v_j^1(t)\boldsymbol{\varphi}_j^1(\boldsymbol{x}; t) + v_j^2(t)\boldsymbol{\varphi}_j^2(\boldsymbol{x}; t). \tag{2.62}$$

where $\boldsymbol{\varphi}_{j1}^1$ and $\boldsymbol{\varphi}_j^2$ again contain the scalar-valued basis function φ_j in respectively their first and second position, and zero in the other. We use linear basis functions for $\varphi_j(\boldsymbol{x}; t)$, centered in the vertex $\boldsymbol{x}_j$. Note that the basis functions depend implicitly on t, since the vertices $\boldsymbol{x}_j$ depend on time. Moreover, the basis functions are Lagrangian, making the weak form (2.60) valid. Substitute (2.61) and (2.62) into (2.60), and set $\boldsymbol{\varphi}$ equal to $\boldsymbol{\varphi}_i^1$ and $\boldsymbol{\varphi}_i^2$ for $i = 1, \ldots, n$. Due to the splitting of $\boldsymbol{\sigma}$ into an elastic and viscous part, we obtain the following system of equations:

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{pmatrix} \rho M \boldsymbol{v}^1 \\ \rho M \boldsymbol{v}^2 \end{pmatrix} + \begin{pmatrix} S_{\text{el}}^{11} & S_{\text{el}}^{12} \\ S_{\text{el}}^{21} & S_{\text{el}}^{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^1 \\ \boldsymbol{u}^2 \end{pmatrix} + \begin{pmatrix} S_{\text{vis}}^{11} & S_{\text{vis}}^{12} \\ S_{\text{vis}}^{21} & S_{\text{vis}}^{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^1 \\ \boldsymbol{v}^2 \end{pmatrix} = \begin{pmatrix} \boldsymbol{g}^1 \\ \boldsymbol{g}^2 \end{pmatrix}, \tag{2.63}$$

where the vectors $\boldsymbol{u}^1$, $\boldsymbol{u}^2$, $\boldsymbol{v}^1$ and $\boldsymbol{v}^2$ contain the coefficients u_j^1, u_j^2, v_j^1 and v_j^2 respectively. The mass matrix M is given by

$$M_{ij} = \int_{\Omega(t)} \varphi_j \varphi_i \,\mathrm{d}\Omega. \tag{2.64}$$

The right-hand side vectors of (2.63) now include both the body force and shear force, they are given by:

$$(\boldsymbol{g}^1)_i = \int_{\Omega(t)} \boldsymbol{g}\cdot\boldsymbol{\varphi}_i^1 \,\mathrm{d}\Omega + \int_{\Gamma_2(t)} \boldsymbol{\tau}\cdot\boldsymbol{\varphi}_i^1 \,\mathrm{d}\Gamma = \int_{\Omega(t)} g^1\varphi_i \,\mathrm{d}\Omega + \int_{\Gamma_2(t)} \tau^1\varphi_i \,\mathrm{d}\Gamma, \tag{2.65}$$

$$(\boldsymbol{g}^2)_i = \int_{\Omega(t)} \boldsymbol{g}\cdot\boldsymbol{\varphi}_i^2 \,\mathrm{d}\Omega + \int_{\Gamma_2(t)} \boldsymbol{\tau}\cdot\boldsymbol{\varphi}_i^2 \,\mathrm{d}\Gamma = \int_{\Omega(t)} g^2\varphi_i \,\mathrm{d}\Omega + \int_{\Gamma_2(t)} \tau^2\varphi_i \,\mathrm{d}\Gamma. \tag{2.66}$$

Moreover, the elements of the S_{el} and S_{vis} blocks are very similar to (2.30)-(2.33):

$$(S_{\text{el}}^{11})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^1 : \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{\varphi}_j^1) \,\mathrm{d}\Omega = \int_{\Omega(t)} (\mu_\varepsilon+\lambda_\varepsilon)\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y} \,\mathrm{d}\Omega, \tag{2.67}$$

$$(S_{\text{el}}^{12})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^1 : \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{\varphi}_j^2) \,\mathrm{d}\Omega = \int_{\Omega(t)} \lambda_\varepsilon\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x} \,\mathrm{d}\Omega, \tag{2.68}$$

$$(S_{\text{el}}^{21})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^2 : \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{\varphi}_j^1) \,\mathrm{d}\Omega = \int_{\Omega(t)} \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \lambda_\varepsilon\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x} \,\mathrm{d}\Omega, \tag{2.69}$$

$$(S_{\text{el}}^{22})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^2 : \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{\varphi}_j^2) \,\mathrm{d}\Omega = \int_{\Omega(t)} \frac{1}{2}\mu_\varepsilon\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + (\mu_\varepsilon+\lambda_\varepsilon)\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y} \,\mathrm{d}\Omega. \tag{2.70}$$

And:

$$(S_{\text{vis}}^{11})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^1 : \boldsymbol{\sigma}_{v}(\boldsymbol{\varphi}_j^1) \,\mathrm{d}\Omega = \int_{\Omega(t)} (\mu_v+\lambda_v)\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + \frac{1}{2}\mu_v\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y} \,\mathrm{d}\Omega, \tag{2.71}$$

$$(S_{\text{vis}}^{12})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^1 : \boldsymbol{\sigma}_{v}(\boldsymbol{\varphi}_j^2) \,\mathrm{d}\Omega = \int_{\Omega(t)} \lambda_v\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \frac{1}{2}\mu_v\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x} \,\mathrm{d}\Omega, \tag{2.72}$$

$$(S_{\text{vis}}^{21})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^2 : \boldsymbol{\sigma}_{v}(\boldsymbol{\varphi}_j^1) \,\mathrm{d}\Omega = \int_{\Omega(t)} \frac{1}{2}\mu_v\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \lambda_v\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x} \,\mathrm{d}\Omega, \tag{2.73}$$

$$(S_{\text{vis}}^{22})_{ij} = \int_{\Omega(t)} \nabla\boldsymbol{\varphi}_i^2 : \boldsymbol{\sigma}_{v}(\boldsymbol{\varphi}_j^2) \,\mathrm{d}\Omega = \int_{\Omega(t)} \frac{1}{2}\mu_v\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + (\mu_v+\lambda_v)\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y} \,\mathrm{d}\Omega. \tag{2.74}$$

Note that we have changed the strain tensor $\boldsymbol{\varepsilon}(\boldsymbol{\varphi})$ in the weak form (2.60) into $\nabla\boldsymbol{\varphi}$. This is valid as per the derivation in (2.20). We can write the system in (2.63) in the even shorter notation

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}(\widetilde{M}\boldsymbol{v}) + S_{\text{el}}\boldsymbol{u} + S_{\text{vis}}\boldsymbol{v} = \boldsymbol{g}, \tag{2.75}$$

where

$$\boldsymbol{u} = \begin{pmatrix}\boldsymbol{u}^1\\ \boldsymbol{u}^2\end{pmatrix}, \quad \boldsymbol{v} = \begin{pmatrix}\boldsymbol{v}^1\\ \boldsymbol{v}^2\end{pmatrix}, \quad \boldsymbol{g} = \begin{pmatrix}\boldsymbol{g}^1\\ \boldsymbol{g}^2\end{pmatrix}, \tag{2.76}$$

and

$$\widetilde{M} = \begin{pmatrix}M & 0\\ 0 & M\end{pmatrix}, \quad S_{\text{el}} = \begin{pmatrix}S_{\text{el}}^{11} & S_{\text{el}}^{12}\\ S_{\text{el}}^{21} & S_{\text{el}}^{22}\end{pmatrix}, \quad S_{\text{vis}} = \begin{pmatrix}S_{\text{vis}}^{11} & S_{\text{vis}}^{12}\\ S_{\text{vis}}^{21} & S_{\text{vis}}^{22}\end{pmatrix}. \tag{2.77}$$

Note that we are 'overloading' the symbols $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{g}$. They denote both a physical quantity and a numerical vector containing unknowns. Their meaning should be clear from the context.

2.3.5 Time Discretization

We use Implicit Euler to discretize (2.75) in time. It is important to note that the matrices M, S_{el}, S_{vis} and the vector $\boldsymbol{g}$ all depend on t, since they all involve integrations over the time-dependent domain $\Omega(t)$ of the time-dependent basis functions. Applying Implicit Euler yields

$$\left(\widetilde{M}^m + \Delta t S_{\text{vis}}^m\right)\boldsymbol{v}^m + \Delta t S_{\text{el}}^m\boldsymbol{u}^m = \widetilde{M}^{m-1}\boldsymbol{v}^{m-1} + \Delta t\boldsymbol{g}^m. \tag{2.78}$$

The superscript denotes a time level. To solve for $\boldsymbol{v}^m$ and $\boldsymbol{u}^m$, we need another equation. This equation is obtained by discretizing the second equation (2.48). In the framework of a moving grid, this turns out to be rather straight-forward.

Lemma 2. *Let $\boldsymbol{u}_i(t)$ be shorthand for $\boldsymbol{u}(\boldsymbol{x}(\boldsymbol{X}_i,t),t)$, then*

$$\frac{D\boldsymbol{u}_i}{Dt} = \frac{\boldsymbol{u}_i(t+\Delta t)-\boldsymbol{u}_i(t)}{\Delta t} + \mathcal{O}(\Delta t).$$

While the statement in Lemma 2 might seem obvious, it is more nuanced than a simple first order approximation of a time-derivative. The movement of the grid is encoded in the displacement vector $\boldsymbol{u}$.

Proof. Use a first order Taylor expansion to expand $\boldsymbol{u}_i(t+\Delta t)$ around $\boldsymbol{u}_i(t)$:

$$\boldsymbol{u}_i(t+\Delta t) = \boldsymbol{u}_i(t) + \Delta t \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{u}_i(t) + \mathcal{O}(\Delta t^2).$$

Let us calculate the time derivative of $\boldsymbol{u}_i$:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{u}_i(t) &= \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{u}(\boldsymbol{x}(\boldsymbol{X}_i,t),t) \\ &= \frac{\partial \boldsymbol{u}}{\partial t}(\boldsymbol{x}(\boldsymbol{X}_i,t),t) + \nabla \boldsymbol{u}(\boldsymbol{x}(\boldsymbol{X}_i,t),t) \cdot \frac{\partial \boldsymbol{x}}{\partial t}(\boldsymbol{X}_i,t) \\ &= \frac{D\boldsymbol{u}}{Dt}(\boldsymbol{x}(\boldsymbol{X}_i,t),t) = \frac{D\boldsymbol{u}_i}{Dt}.\end{aligned}$$

By rearranging terms and dividing by Δt one finds the desired expression. □

Using Lemma 2, we again discretize (2.48) by

$$\boldsymbol{u}^m - \boldsymbol{u}^{m-1} = \Delta t \boldsymbol{v}^m, \tag{2.79}$$

where we again use Implicit Euler. Joining equations (2.78) and (2.79) into a single system yields

$$\begin{pmatrix} \widetilde{M}^m + \Delta t S^m_{\mathrm{vis}} & \Delta t S^m_{\mathrm{el}} \\ -\Delta t I & I \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^m \\ \boldsymbol{u}^m \end{pmatrix} = \begin{pmatrix} \widetilde{M}^{m-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^{m-1} \\ \boldsymbol{u}^{m-1} \end{pmatrix} + \Delta t \begin{pmatrix} \boldsymbol{g}^m \\ \boldsymbol{0} \end{pmatrix}. \tag{2.80}$$

Since the grid at time level m depends on $\boldsymbol{u}^m$, we cannot solve (2.80) directly for $\boldsymbol{u}^m$ and $\boldsymbol{v}^m$. Instead, we use an iterative scheme. Note that the system in (2.80) can be abstractly written as

$$A(z)z = c + b(z), \tag{2.81}$$

where z is the unknown vector, $A(z)$ is the left-hand side matrix, c is a constant vector corresponding to the solution at time t, and $b(z)$ is the right-hand side vector containing $\boldsymbol{g}^m$. We approximate the solution to (2.81) by generating a sequence (z_k) using the following update rule:

$$A(z_k)z_{k+1} = c + b(z_k). \tag{2.82}$$

Where z_0 is equal to the unkowns at time level $m-1$. If we translate this scheme back to the original variables, we get

$$\begin{pmatrix} \widetilde{M}_k + \Delta t (S_{\mathrm{vis}})_k & \Delta t (S_{\mathrm{el}})_k \\ -\Delta t I & I \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_{k+1} \\ \boldsymbol{u}_{k+1} \end{pmatrix} = \begin{pmatrix} \widetilde{M}^{m-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^{m-1} \\ \boldsymbol{u}^{m-1} \end{pmatrix} + \Delta t \begin{pmatrix} \boldsymbol{g}_k \\ \boldsymbol{0} \end{pmatrix}. \tag{2.83}$$

Where the subscript k means the matrix or vector has been calculated on the grid based on $\boldsymbol{u}_k$. In words, one iteration of the scheme amounts to:

- Create new grid based on $\boldsymbol{u}_k$.
- Calculate matrices $\widetilde{M}_k$, $(S_{\mathrm{vis}})_k$, $(S_{\mathrm{el}})_k$ and vector $\boldsymbol{g}_k$ on the new grid.
- Solve (2.83) for $\boldsymbol{u}_{k+1}$ (and $\boldsymbol{v}_{k+1}$).

If the iterative process converges, we have found a fixed point of (2.83), and thus a solution to (2.80). In practice, we say that the scheme has converged if the relative difference between two consecutive iterates is sufficiently small. That is, if

$$\frac{\|\boldsymbol{u}_{k+1} - \boldsymbol{u}_k\|}{\|\boldsymbol{u}_k\|} + \frac{\|\boldsymbol{v}_{k+1} - \boldsymbol{v}_k\|}{\|\boldsymbol{v}_k\|} < \delta, \tag{2.84}$$

for some small $\delta > 0$.

2.3.6 Numerical Experiments

We run the Finite Element model using the parameters $\rho = 1$, $\mu_\varepsilon = \lambda_\varepsilon = \mu_v = \lambda_v = 1$. The initial domain is the unit square: $\Omega(0) = (0,1)^2$. We apply the body force $\boldsymbol{g} = (0, -0.1)^\top$. On the left border ($x = 0$) a homogeneous Dirichlet boundary condition is imposed; this border is fixed. On the remaining boundaries, a homogeneous Neumann boundary condition is imposed, corresponding to $\boldsymbol{\tau} = \mathbf{0}$. The solution is expected to converge to the stationary solution, which is shown in Figure 2.5. Note that the stationary solution has been computed using the elastic Finite Element model from section 2.2.

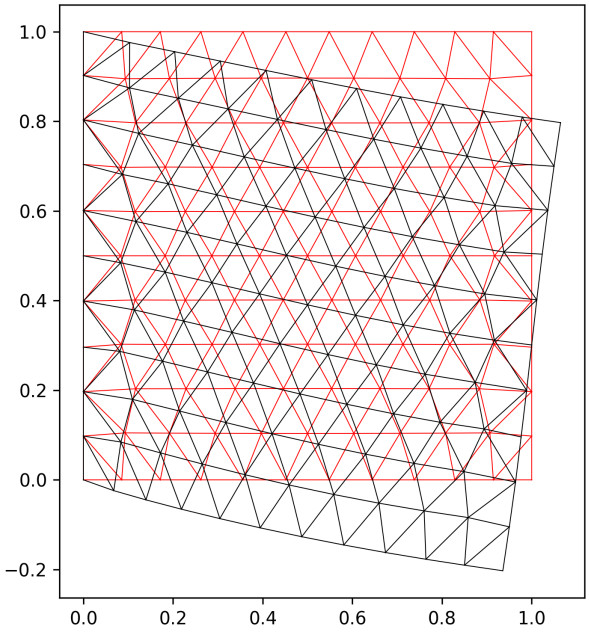
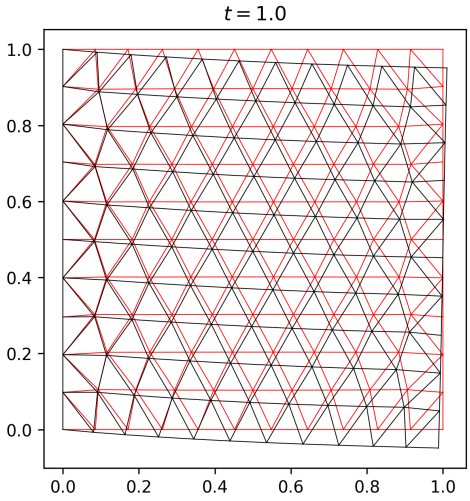
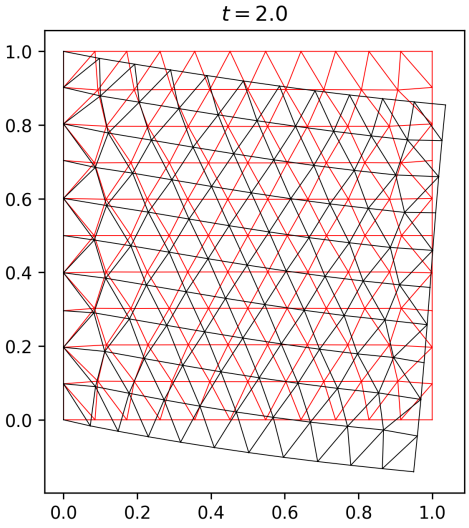


Figure 2.5: Stationary solution. A meshwidth of $h = 0.1$ has been used.

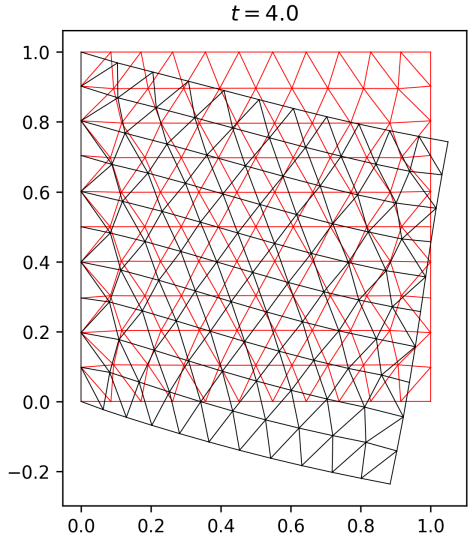
We solve the viscoelastic system from $t = 0$ to $t = 8$ using a stepsize of $\Delta t = 0.05$. The same (initial) meshwidth of $h = 0.1$ is used. Figures 2.6a - 2.6d show the intermediate finite element solutions. Note that in Figure 2.6c the mesh has passed the equilibrium state. Due to the elastic nature of the material, it will again bounce back, as can be seen in Figure 2.6d. This is a result incorporating the viscous term in the force-balance equation.



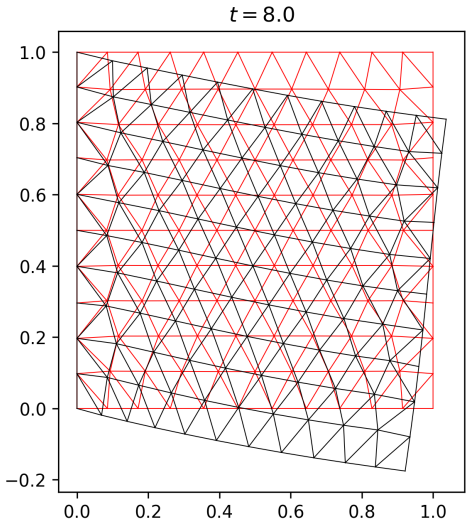
(a) Finite element solution at $t = 1.0$.



(b) Finite element solution at $t = 2.0$.



(c) Finite element solution at $t = 4.0$.



(d) Finite element solution at $t = 8.0$.

Figure 2.6: Plots showing the finite element solution at various points in time.

2.4 Morphoelasticity

Using morphoelasticity allows us to model elastic growth. The idea as introduced in [12] is to decompose the deformation tensor $\boldsymbol{F}$ as defined in (2.45) into an elastic part $\boldsymbol{F}^e$ and plastic part $\boldsymbol{F}^p$:

$$\boldsymbol{F} = \boldsymbol{F}^e \boldsymbol{F}^p. \tag{2.85}$$

An intuitive way to look at this decomposition is to consider the corresponding mappings that map coordinates onto each other. Between the initial state $\boldsymbol{X}$ and current state $\boldsymbol{x}$, a stress-free state $\boldsymbol{z}$ is introduced. In the stress free state, only the plastic deformation is applied. The decomposition can then be represented as

$$\boldsymbol{F} : \boldsymbol{X} \xrightarrow{\boldsymbol{F}^p} \boldsymbol{z} \xrightarrow{\boldsymbol{F}^e} \boldsymbol{x}. \tag{2.86}$$

The stress-free state is a virtual configuration; the mapping $\boldsymbol{F}^p$ need not even be continuous. In [8], the theory of morphoelasticity is used to obtain an evolution equation for the strain tensor $\boldsymbol{\varepsilon}$. It is important to note that the structure of the problem now becomes completely different compared to viscoelasticity. In the latter framework, $\boldsymbol{\varepsilon}$ directly related to the displacement $\boldsymbol{u}$, allowing us to construct a model that computes $\boldsymbol{u}$ directly. In the framework of morphoelasticity, $\boldsymbol{u}$ will no longer be explicitly featured in any

of the equations. Instead, the model only allows us to compute $\boldsymbol{\varepsilon}$ and the displacement velocity $\boldsymbol{v}$, from which $\boldsymbol{u}$ can then be determined in a post-processing step. We will not reproduce the derivation of the strain evolution equation from [8], as it involves many pages of algebraic manipulation. Together with (2.36), we obtain the following morphoelastic system of equations:

$$\rho\left(\frac{D\boldsymbol{v}}{Dt} + (\nabla\cdot\boldsymbol{v})\boldsymbol{v}\right) - \nabla\cdot\boldsymbol{\sigma}(\boldsymbol{v},\boldsymbol{\varepsilon}) = \boldsymbol{g} \tag{2.87}$$

$$\frac{D\boldsymbol{\varepsilon}}{Dt} + \boldsymbol{\varepsilon}\mathrm{skw}(\nabla\boldsymbol{v}) - \mathrm{skw}(\nabla\boldsymbol{v})\boldsymbol{\varepsilon} + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\mathrm{sym}(\nabla\boldsymbol{v}) = -\boldsymbol{G}. \tag{2.88}$$

The stress tensor is again given by an elastic and viscous part such as in (2.37), but the elastic part now depends directly on $\boldsymbol{\varepsilon}$ instead of $\boldsymbol{u}$:

$$\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) = \mu_\varepsilon\boldsymbol{\varepsilon} + \lambda_\varepsilon\mathrm{tr}(\boldsymbol{\varepsilon})\boldsymbol{I}. \tag{2.89}$$

Note that to obtain the updated locations of the grid nodes, one must still compute $\boldsymbol{u}$. It can be extracted from $\boldsymbol{v}$. Like in the previous section, part of the boundary is fixed by setting $\boldsymbol{v} = \boldsymbol{0}$ on $\Gamma_1(t)$, and on the other part the shear force is prescribed:

$$\boldsymbol{\sigma}\cdot\boldsymbol{n} = \boldsymbol{\tau}, \quad \text{on } \Gamma_2(t). \tag{2.90}$$

To reduce the number of unknowns by one, we shall first prove the following lemma.

Lemma 3 (Daan Smits [16])**.** *If $\boldsymbol{G}$ is symmetric for all $t \geqslant 0$ and $\boldsymbol{\varepsilon}$ is symmetric at $t = 0$, then $\boldsymbol{\varepsilon}$ is symmetric for all $t \geqslant 0$.*

Proof. Consider the transpose of equation (2.88):

$$\frac{D\boldsymbol{\varepsilon}^\top}{Dt} + \mathrm{skw}(\nabla\boldsymbol{v})^\top\boldsymbol{\varepsilon}^\top - \boldsymbol{\varepsilon}^\top\mathrm{skw}(\nabla\boldsymbol{v})^\top + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\mathrm{sym}(\nabla\boldsymbol{v})^\top = -\boldsymbol{G}^\top.$$

Using the facts that $\boldsymbol{G}^\top = \boldsymbol{G}$, $\mathrm{skw}(\nabla\boldsymbol{v})^\top = -\mathrm{skw}(\nabla\boldsymbol{v})$ and $\mathrm{sym}(\nabla\boldsymbol{v})^\top = \mathrm{sym}(\nabla\boldsymbol{v})$, we get

$$\frac{D\boldsymbol{\varepsilon}^\top}{Dt} + \boldsymbol{\varepsilon}^\top\mathrm{skw}(\nabla\boldsymbol{v}) - \mathrm{skw}(\nabla\boldsymbol{v})\boldsymbol{\varepsilon}^\top + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\mathrm{sym}(\nabla\boldsymbol{v}) = -\boldsymbol{G}.$$

Subtract the above equation from (2.88) to obtain

$$\frac{D(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^\top)}{Dt} + (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^\top)\mathrm{skw}(\nabla\boldsymbol{v}) - \mathrm{skw}(\nabla\boldsymbol{v})(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^\top) = \boldsymbol{0}.$$

Thus, whenever $\boldsymbol{\varepsilon}$ is symmetric, we have

$$\frac{D(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^\top)}{Dt} = \boldsymbol{0}.$$

Since $\boldsymbol{\varepsilon}$ is symmetric at $t = 0$, it will be symmetric for all $t \geqslant 0$. □

In the numerical experiments, we will set $\boldsymbol{G} = \alpha\boldsymbol{\varepsilon}$. In this case Lemma 3 holds whenever $\boldsymbol{\varepsilon}$ is symmetric at $t = 0$. We always set $\boldsymbol{\varepsilon}|_{t=0} = \boldsymbol{0}$. From now on, we will assume the conditions of Lemma 3 are met. Since $\varepsilon^{12} = \varepsilon^{21}$, we are left with the 5 unknowns v^1, v^2 and $\varepsilon^{11}, \varepsilon^{12}, \varepsilon^{22}$.

2.4.1 Weak Formulation

Velocity Evolution Equation

We first consider the weak form of (2.87). Note that we can repeat the steps in section 2.3.3 up until equation (2.60). We must now however split the stress-tensor into a viscous part and strain part to obtain the following weak form:

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)}\boldsymbol{v}\cdot\boldsymbol{\varphi}\,\mathrm{d}\Omega + \int_{\Omega(t)}(\boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) + \boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon})) : \nabla\boldsymbol{\varphi}\,\mathrm{d}\Omega = \int_{\Omega(t)}\boldsymbol{g}\cdot\boldsymbol{\varphi}\,\mathrm{d}\Omega + \int_{\Gamma(t)}\boldsymbol{\tau}\cdot\boldsymbol{\varphi}\,\mathrm{d}\Omega. \tag{2.91}$$

The integral of $\boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) : \nabla\boldsymbol{\varphi}$ will lead to the same Galerkin equations as in section 2.3.4. Therefore we will focus on the part involving $\boldsymbol{\varepsilon}$. Let φ be a scalar-valued Lagrangian basis function. If we first let $\boldsymbol{\varphi} = (\varphi, 0)^\top$, then

$$\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) : \nabla\boldsymbol{\varphi} = \left((\mu_\varepsilon + \lambda_\varepsilon)\varepsilon^{11} + \lambda_\varepsilon\varepsilon^{22}\right)\frac{\partial\varphi}{\partial x} + \mu_\varepsilon\varepsilon^{12}\frac{\partial\varphi}{\partial y}. \tag{2.92}$$

Now let $\boldsymbol{\varphi} = (0, \varphi)^\top$, then:

$$\boldsymbol{\sigma}_{\text{el}}(\boldsymbol{\varepsilon}) : \nabla\boldsymbol{\varphi} = \mu_\varepsilon \varepsilon^{12} \frac{\partial \varphi}{\partial x} + \left(\lambda_\varepsilon \varepsilon^{11} + (\mu_\varepsilon + \lambda_\varepsilon)\varepsilon^{22}\right) \frac{\partial \varphi}{\partial y}. \tag{2.93}$$

By considering these expressions and using (2.71) - (2.74), we get the following weak form for the equation for v_1:

$$\begin{aligned}\rho \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} v^1 \varphi \,\mathrm{d}\Omega + \int_{\Omega(t)} \left((\mu_v + \lambda_v) \frac{\partial v^1}{\partial x} + \lambda_v \frac{\partial v^2}{\partial y} + (\mu_\varepsilon + \lambda_\varepsilon)\varepsilon^{11} + \lambda_\varepsilon \varepsilon^{22} \right) \frac{\partial \varphi}{\partial x} \\ + \left(\frac{1}{2}\mu_v \frac{\partial v^1}{\partial y} + \frac{1}{2}\mu_v \frac{\partial v^2}{\partial x} + \mu_\varepsilon \varepsilon^{12} \right) \frac{\partial \varphi}{\partial y} \,\mathrm{d}\Omega = \int_{\Omega(t)} g^1 \varphi \,\mathrm{d}\Omega + \int_{\Gamma(t)} \tau^1 \varphi \,\mathrm{d}\Gamma,\end{aligned} \tag{2.94}$$

and for the equation for v_2:

$$\begin{aligned}\rho \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} v^2 \varphi \,\mathrm{d}\Omega + \int_{\Omega(t)} \left(\frac{1}{2}\mu_v \frac{\partial v^1}{\partial y} + \frac{1}{2}\mu_v \frac{\partial v^2}{\partial x} + \mu_\varepsilon \varepsilon^{12} \right) \frac{\partial \varphi}{\partial x} \\ + \left(\lambda_v \frac{\partial v^1}{\partial x} + (\mu_v + \lambda_v) \frac{\partial v^2}{\partial y} + \lambda_\varepsilon \varepsilon^{11} + (\mu_\varepsilon + \lambda_\varepsilon)\varepsilon^{22} \right) \frac{\partial \varphi}{\partial y} \,\mathrm{d}\Omega = \int_{\Omega(t)} g^2 \varphi \,\mathrm{d}\Omega + \int_{\Gamma(t)} \tau^2 \varphi \,\mathrm{d}\Gamma,\end{aligned} \tag{2.95}$$

Strain Evolution Equation
We write out the strain evolution equation (2.88) component-wise:

$$\frac{D\varepsilon^{11}}{Dt} + \varepsilon^{21} \left(\frac{\partial v^2}{\partial x} - \frac{\partial v^1}{\partial y} \right) + (\varepsilon^{11} + \varepsilon^{22} - 1) \frac{\partial v^1}{\partial x} = -G^{11}, \tag{2.96}$$

$$\frac{D\varepsilon^{12}}{Dt} + \frac{1}{2}(\varepsilon^{11} - \varepsilon^{22}) \left(\frac{\partial v^1}{\partial y} - \frac{\partial v^2}{\partial x} \right) + \frac{1}{2}(\varepsilon^{11} + \varepsilon^{22} - 1) \left(\frac{\partial v^2}{\partial x} + \frac{\partial v^1}{\partial y} \right) = -G^{12}, \tag{2.97}$$

$$\frac{D\varepsilon^{22}}{Dt} + \varepsilon^{21} \left(\frac{\partial v^1}{\partial y} - \frac{\partial v^2}{\partial x} \right) + (\varepsilon^{11} + \varepsilon^{22} - 1) \frac{\partial v^2}{\partial y} = -G^{22}. \tag{2.98}$$

Before we multiply these equations by a test function and integrate them over $\Omega(t)$, we add a term involving $\nabla \cdot \boldsymbol{v}$ to both sides. This is done in order to use Theorem 3. After rearranging some terms, we find that (2.96) - (2.98) are equivalent to:

$$\left(\frac{D\varepsilon^{11}}{Dt} + (\nabla \cdot \boldsymbol{v})\varepsilon^{11} \right) - \varepsilon^{11} \frac{\partial v^2}{\partial y} + \varepsilon^{12} \left(\frac{\partial v^2}{\partial x} - \frac{\partial v^1}{\partial y} \right) + \varepsilon^{22} \frac{\partial v^1}{\partial x} = \frac{\partial v^1}{\partial x} - G^{11}, \tag{2.99}$$

$$\left(\frac{D\varepsilon^{12}}{Dt} + (\nabla \cdot \boldsymbol{v})\varepsilon^{12} \right) + \varepsilon^{11} \frac{\partial v^1}{\partial y} - \varepsilon^{12} \left(\frac{\partial v^1}{\partial x} + \frac{\partial v^2}{\partial y} \right) + \varepsilon^{22} \frac{\partial v^2}{\partial x} = \frac{1}{2} \left(\frac{\partial v^2}{\partial x} + \frac{\partial v^1}{\partial y} \right) - G^{12}, \tag{2.100}$$

$$\left(\frac{D\varepsilon^{22}}{Dt} + (\nabla \cdot \boldsymbol{v})\varepsilon^{22} \right) + \varepsilon^{11} \frac{\partial v^2}{\partial y} - \varepsilon^{12} \left(\frac{\partial v^2}{\partial x} - \frac{\partial v^1}{\partial y} \right) - \varepsilon^{22} \frac{\partial v^1}{\partial x} = \frac{\partial v^2}{\partial y} - G^{22}. \tag{2.101}$$

Let φ be a Lagrangian test function, then by Theorem 3 we find the following weak forms:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \varepsilon^{11} \varphi \,\mathrm{d}\Omega + \int_{\Omega(t)} \left(-\varepsilon^{11} \frac{\partial v^2}{\partial y} + \varepsilon^{12} \left(\frac{\partial v^2}{\partial x} - \frac{\partial v^1}{\partial y} \right) + \varepsilon^{22} \frac{\partial v^1}{\partial x} \right) \varphi \,\mathrm{d}\Omega \\ = \int_{\Omega(t)} \left(\frac{\partial v^1}{\partial x} - G^{11} \right) \varphi \,\mathrm{d}\Omega,\end{aligned} \tag{2.102}$$

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \varepsilon^{12} \varphi \,\mathrm{d}\Omega + \int_{\Omega(t)} \left(\varepsilon^{11} \frac{\partial v^1}{\partial y} - \varepsilon^{12} \left(\frac{\partial v^1}{\partial x} + \frac{\partial v^2}{\partial y} \right) + \varepsilon^{22} \frac{\partial v^2}{\partial x} \right) \varphi \,\mathrm{d}\Omega \\ = \int_{\Omega(t)} \left(\frac{1}{2} \left(\frac{\partial v^2}{\partial x} + \frac{\partial v^1}{\partial y} \right) - G^{12} \right) \varphi \,\mathrm{d}\Omega,\end{aligned} \tag{2.103}$$

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \varepsilon^{22} \varphi \,\mathrm{d}\Omega + \int_{\Omega(t)} \left(\varepsilon^{11} \frac{\partial v^2}{\partial y} - \varepsilon^{12} \left(\frac{\partial v^2}{\partial x} - \frac{\partial v^1}{\partial y} \right) - \varepsilon^{22} \frac{\partial v^1}{\partial x} \right) \varphi \,\mathrm{d}\Omega \\ = \int_{\Omega(t)} \left(\frac{\partial v^2}{\partial y} - G^{22} \right) \varphi \,\mathrm{d}\Omega,\end{aligned} \tag{2.104}$$

Another way to obtain these equations is to multiply (2.88) with a symmetric tensor-valued test field $\boldsymbol{\zeta}$, and integrate over $\Omega(t)$. To use Theorem 3, first add and subtract $(\nabla \cdot \boldsymbol{v})\boldsymbol{\varepsilon}$. We then get the weak form

$$\begin{aligned}\int_{\Omega(t)} \left(\frac{D\boldsymbol{\varepsilon}}{Dt} + (\nabla \cdot \boldsymbol{v})\boldsymbol{\varepsilon}\right) : \boldsymbol{\zeta} \, \mathrm{d}\Omega \\ + \int_{\Omega(t)} \Big(\boldsymbol{\varepsilon}\mathrm{skw}(\nabla \boldsymbol{v}) - \mathrm{skw}(\nabla \boldsymbol{v})\boldsymbol{\varepsilon} + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\mathrm{sym}(\nabla \boldsymbol{v}) - (\nabla \cdot \boldsymbol{v})\boldsymbol{\varepsilon}\Big) : \boldsymbol{\zeta} \, \mathrm{d}\Omega \\ = -\int_{\Omega(t)} \boldsymbol{G} : \boldsymbol{\zeta} \, \mathrm{d}\Omega. \quad (2.105)\end{aligned}$$

Applying Theorem 3 yields

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)} \boldsymbol{\varepsilon} : \boldsymbol{\zeta} \, \mathrm{d}\Omega \\ + \int_{\Omega(t)} \Big(\boldsymbol{\varepsilon}\mathrm{skw}(\nabla \boldsymbol{v}) - \mathrm{skw}(\nabla \boldsymbol{v})\boldsymbol{\varepsilon} + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\mathrm{sym}(\nabla \boldsymbol{v}) - (\nabla \cdot \boldsymbol{v})\boldsymbol{\varepsilon}\Big) : \boldsymbol{\zeta} \, \mathrm{d}\Omega \\ = -\int_{\Omega(t)} \boldsymbol{G} : \boldsymbol{\zeta} \, \mathrm{d}\Omega. \quad (2.106)\end{aligned}$$

This step is valid provided $\boldsymbol{\zeta}$ is a Lagrangian test field. Setting $\boldsymbol{\zeta}$ equal to

$$\begin{pmatrix} \varphi & 0 \\ 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & \varphi \\ 0 & 0 \end{pmatrix}, \quad \text{and} \quad \begin{pmatrix} 0 & 0 \\ 0 & \varphi \end{pmatrix},$$

yields equations (2.102), (2.103) and (2.104) respectively. To further motivative this, the tensor in the middle term of equation (2.106) is computed component-wise:

$$\begin{aligned}&\boldsymbol{\varepsilon}\mathrm{skw}(\nabla \boldsymbol{v}) - \mathrm{skw}(\nabla \boldsymbol{v})\boldsymbol{\varepsilon} + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\mathrm{sym}(\nabla \boldsymbol{v}) - (\nabla \cdot \boldsymbol{v})\boldsymbol{\varepsilon} \\ &= \begin{pmatrix} -\varepsilon^{11}\dfrac{\partial v^2}{\partial y} + \varepsilon^{12}\left(\dfrac{\partial v^2}{\partial x} - \dfrac{\partial v^1}{\partial y}\right) + \varepsilon^{22}\dfrac{\partial v^1}{\partial x} & \varepsilon^{11}\dfrac{\partial v^1}{\partial y} - \varepsilon^{12}\left(\dfrac{\partial v_1}{\partial x} + \dfrac{\partial v^2}{\partial y}\right) + \varepsilon^{22}\dfrac{\partial v^2}{\partial x} \\ \varepsilon^{11}\dfrac{\partial v^1}{\partial y} - \varepsilon^{12}\left(\dfrac{\partial v^1}{\partial x} + \dfrac{\partial v^2}{\partial y}\right) + \varepsilon^{22}\dfrac{\partial v^2}{\partial x} & \varepsilon^{11}\dfrac{\partial v^2}{\partial y} - \varepsilon^{12}\left(\dfrac{\partial v^2}{\partial x} - \dfrac{\partial v^1}{\partial y}\right) - \varepsilon^{22}\dfrac{\partial v^1}{\partial x} \end{pmatrix} \\ &\qquad - \begin{pmatrix} \dfrac{\partial v^1}{\partial x} & \dfrac{1}{2}\left(\dfrac{\partial v^1}{\partial y} + \dfrac{\partial v^2}{\partial x}\right) \\ \dfrac{1}{2}\left(\dfrac{\partial v^1}{\partial y} + \dfrac{\partial v^2}{\partial x}\right) & \dfrac{\partial v^2}{\partial y} \end{pmatrix}. \quad (2.107)\end{aligned}$$

One can then immediately see that taking the double dot product with one of the test tensors above leads to equations (2.102) - (2.104).

2.4.2 Galerkin Equations

We derive the Galerkin equations corresponding to weak forms (2.94), (2.95) and (2.102), (2.103), (2.104). Let φ_j be the linear basis function centered in grid node j. The finite element solutions are then of the form

$$v^1 = \sum_{j=1}^{n} v_j^1 \varphi_j, \quad v^2 = \sum_{j=1}^{n} v_j^2 \varphi_j, \quad \varepsilon^{11} = \sum_{j=1}^{n} \varepsilon_j^{11} \varphi_j, \quad \varepsilon^{12} = \sum_{j=1}^{n} \varepsilon_j^{12} \varphi_j, \quad \varepsilon^{22} = \sum_{j=1}^{n} \varepsilon_j^{22} \varphi_j, \qquad (2.108)$$

The coefficients $v_j^1, \ldots, \varepsilon_j^{22}$ depend only on time, whereas the basis functions depend on space and time; $\varphi_j = \varphi_j(\boldsymbol{x}; t)$. Substitute the finite element solutions into the weak forms, and set $\varphi = \varphi_i$ for $i = 1, \ldots, n$. Then each weak form will yield n equations. Note that the equations derived from (2.102), (2.103) and (2.104) will contain a non-linear part, due to the products between ε-components and spatial derivatives

of the components of $\boldsymbol{v}$. The resulting non-linear system is given in block matrix form by:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix}\rho M\boldsymbol{v}^1\\ \rho M\boldsymbol{v}^2\\ M\boldsymbol{\varepsilon}^{11}\\ M\boldsymbol{\varepsilon}^{12}\\ M\boldsymbol{\varepsilon}^{22}\end{pmatrix}+\begin{pmatrix} S^{11}_{\text{vis}} & S^{12}_{\text{vis}} & (\mu_\varepsilon+\lambda_\varepsilon)B_x & \mu_\varepsilon B_y & \lambda_\varepsilon B_x\\ S^{21}_{\text{vis}} & S^{22}_{\text{vis}} & \lambda_\varepsilon B_y & \mu_\varepsilon B_x & (\mu_\varepsilon+\lambda_\varepsilon)B_y\\ -B_x^\top & 0 & 0 & 0 & 0\\ -\frac{1}{2}B_y^\top & -\frac{1}{2}B_x^\top & 0 & 0 & 0\\ 0 & -B_y^\top & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}\boldsymbol{v}^1\\ \boldsymbol{v}^2\\ \boldsymbol{\varepsilon}^{11}\\ \boldsymbol{\varepsilon}^{12}\\ \boldsymbol{\varepsilon}^{22}\end{pmatrix}+\begin{pmatrix}\mathbf{0}\\ \mathbf{0}\\ \boldsymbol{N}^{11}(\boldsymbol{z})\\ \boldsymbol{N}^{12}(\boldsymbol{z})\\ -\boldsymbol{N}^{11}(\boldsymbol{z})\end{pmatrix}$$
$$=\begin{pmatrix}\boldsymbol{g}^1+\boldsymbol{\tau}^1\\ \boldsymbol{g}^2+\boldsymbol{\tau}^2\\ -\boldsymbol{G}^{11}\\ -\boldsymbol{G}^{12}\\ -\boldsymbol{G}^{22}\end{pmatrix}. \tag{2.109}$$

Here $\boldsymbol{v}^1=(v_1^1,\dots,v_n^1)^\top$ etc, and the vector $\boldsymbol{z}$ contains all the unknowns: $\boldsymbol{z}=(\boldsymbol{v}^1,\boldsymbol{v}^2,\boldsymbol{\varepsilon}^{11},\boldsymbol{\varepsilon}^{12},\boldsymbol{\varepsilon}^{22})^\top$. The components of the matrices S^{11}_{vis}, S^{12}_{vis}, S^{21}_{vis} and S^{22}_{vis} are given by (2.71), (2.72), (2.73), and (2.74) respectively. Similarly, the right-hand side vectors $\boldsymbol{g}^1$ and $\boldsymbol{g}^2$ are given by (2.65) and (2.66) respectively. The mass matrix M is again given by:

$$M_{ij}=\int_{\Omega(t)}\varphi_i\varphi_j\,\mathrm{d}\Omega. \tag{2.110}$$

The components of the matrices B_x and B_y are given by:

$$(B_x)_{ij}=\int_{\Omega(t)}\varphi_j\frac{\partial\varphi_i}{\partial x}\,\mathrm{d}\Omega, \tag{2.111}$$

$$(B_y)_{ij}=\int_{\Omega(t)}\varphi_j\frac{\partial\varphi_i}{\partial y}\,\mathrm{d}\Omega. \tag{2.112}$$

The non-linear functionals $\boldsymbol{N}^{11}$ and $\boldsymbol{N}^{12}$ are defined component-wise by:

$$N_i^{11}(\boldsymbol{z})=\sum_{j=1}^n\sum_{k=1}^n\left(-N^y_{ijk}\varepsilon_j^{11}v_k^2-N^y_{ijk}\varepsilon_j^{12}v_k^1+N^x_{ijk}\varepsilon_j^{12}v_k^2+N^x_{ijk}\varepsilon_j^{22}v_k^1\right), \tag{2.113}$$

$$N_i^{12}(\boldsymbol{z})=\sum_{j=1}^n\sum_{k=1}^n\left(N^y_{ijk}\varepsilon_j^{11}v_k^1-N^x_{ijk}\varepsilon_j^{12}v_k^1-N^y_{ijk}\varepsilon_j^{12}v_k^2+N^x_{ijk}\varepsilon_j^{22}v_k^2\right), \tag{2.114}$$

where

$$N^x_{ijk}=\int_{\Omega(t)}\varphi_i\varphi_j\frac{\partial\varphi_k}{\partial x}\,\mathrm{d}\Omega, \tag{2.115}$$

$$N^y_{ijk}=\int_{\Omega(t)}\varphi_i\varphi_j\frac{\partial\varphi_k}{\partial y}\,\mathrm{d}\Omega. \tag{2.116}$$

Finally, the right-hand side vectors $\boldsymbol{G}^{11}$, $\boldsymbol{G}^{12}$ and $\boldsymbol{G}^{22}$ are given by:

$$G_i^{11}=\int_{\Omega(t)}\varphi_iG^{11}\,\mathrm{d}\Omega,\quad G_i^{12}=\int_{\Omega(t)}\varphi_iG^{12}\,\mathrm{d}\Omega,\quad G_i^{22}=\int_{\Omega(t)}\varphi_iG^{22}\,\mathrm{d}\Omega,\quad i,j,k\in\{1,2,3\}. \tag{2.117}$$

The element vectors and matrices of all objects found in system (2.109) are found in Appendix A.2. To model permanent deformations, we let the growth tensor $\boldsymbol{G}$ be proportional to the strain tensor $\boldsymbol{\varepsilon}$:

$$\boldsymbol{G}=\alpha\boldsymbol{\varepsilon}, \tag{2.118}$$

for some proportionality constant $\alpha \geqslant 0$. In this case we have $\boldsymbol{G}^{11} = \alpha M \boldsymbol{\varepsilon}^{11}$, $\boldsymbol{G}^{12} = \alpha M \boldsymbol{\varepsilon}^{12}$ and $\boldsymbol{G}^{22} = \alpha M \boldsymbol{\varepsilon}^{22}$. Thus, system (2.109) becomes:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} \rho M \boldsymbol{v}^1 \\ \rho M \boldsymbol{v}^2 \\ M\boldsymbol{\varepsilon}^{11} \\ M\boldsymbol{\varepsilon}^{12} \\ M\boldsymbol{\varepsilon}^{22} \end{pmatrix} + \begin{pmatrix} S^{11}_{\mathrm{vis}} & S^{12}_{\mathrm{vis}} & (\mu_\varepsilon + \lambda_\varepsilon)B_x & \mu_\varepsilon B_y & \lambda_\varepsilon B_x \\ S^{21}_{\mathrm{vis}} & S^{22}_{\mathrm{vis}} & \lambda_\varepsilon B_y & \mu_\varepsilon B_x & (\mu_\varepsilon + \lambda_\varepsilon)B_y \\ -B_x^\top & 0 & \alpha M & 0 & 0 \\ -\frac{1}{2}B_y^\top & -\frac{1}{2}B_x^\top & 0 & \alpha M & 0 \\ 0 & -B_y^\top & 0 & 0 & \alpha M \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^1 \\ \boldsymbol{v}^2 \\ \boldsymbol{\varepsilon}^{11} \\ \boldsymbol{\varepsilon}^{12} \\ \boldsymbol{\varepsilon}^{22} \end{pmatrix} + \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{N}^{11}(\boldsymbol{z}) \\ \boldsymbol{N}^{12}(\boldsymbol{z}) \\ -\boldsymbol{N}^{11}(\boldsymbol{z}) \end{pmatrix} = \begin{pmatrix} \boldsymbol{g}^1 + \boldsymbol{\tau}^1 \\ \boldsymbol{g}^2 + \boldsymbol{\tau}^2 \\ -\boldsymbol{G}^{11} \\ -\boldsymbol{G}^{12} \\ -\boldsymbol{G}^{22} \end{pmatrix}. \tag{2.119}$$

It will become useful to write the above system in a more compact form. To this end, we define the block vectors $\boldsymbol{v} = (\boldsymbol{v}^1, \boldsymbol{v}^2)^\top$ and $\boldsymbol{\varepsilon} = (\boldsymbol{\varepsilon}^{11}, \boldsymbol{\varepsilon}^{21}, \boldsymbol{\varepsilon}^{22})^\top$. Then system (2.119) can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} \rho M_v \boldsymbol{v} \\ M_\varepsilon \boldsymbol{\varepsilon} \end{pmatrix} + \begin{pmatrix} S_{\mathrm{vis}} & S_{\mathrm{el}} \\ -B^\top & \alpha M_\varepsilon \end{pmatrix} \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{\varepsilon} \end{pmatrix} + \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{N}(\boldsymbol{v}, \boldsymbol{\varepsilon}) \end{pmatrix} = \begin{pmatrix} \boldsymbol{g} + \boldsymbol{\tau} \\ \boldsymbol{G} \end{pmatrix}. \tag{2.120}$$

Here M_v and M_ε the block matrices with respectively 2 and 3 times the mass matrix M on its block-diagonal. The other block matrices and vectors should be clear from the structure of the system when compared to (2.119). The per-element contributions, or element matrices/vectors, of all objects found in this section are computed in A.2.

2.4.3 Time Discretization

We use the following short notation to denote (2.109) or (2.119):

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\widetilde{M}\boldsymbol{z}\right) + A\boldsymbol{z} + \boldsymbol{N}(\boldsymbol{z}) = \boldsymbol{f}. \tag{2.121}$$

Note that $\widetilde{M}$, A, $\boldsymbol{N}(\cdot)$ and $\boldsymbol{f}$ all depend on $\boldsymbol{z}$ implicitly, because they involve integrations over the current domain $\Omega(t)$. We use the implicit Euler timestepping method. Applying this method to (2.121) yields the following update equation:

$$\left(\widetilde{M}^m + \Delta t A^m\right)\boldsymbol{z}^m + \Delta t \boldsymbol{N}^m(\boldsymbol{z}^m) = \widetilde{M}^{m-1}\boldsymbol{z}^{m-1} + \Delta t \boldsymbol{f}^m. \tag{2.122}$$

The superscript denotes a time-level. In the case of the coefficient matrices and right-hand side vector, the time-level is shown explicitly to emphasize that these objects depend on the grid. Note that the grid at time-level m can be computed using $\boldsymbol{v}^m$ like in Section 2.3.5. Given $\boldsymbol{z}^{m-1}$, to obtain $\boldsymbol{z}^m$, we must solve an equation of the form

$$Q[\zeta]\zeta + q[\zeta](\zeta) = \widetilde{f}[\zeta] \tag{2.123}$$

for $\zeta = \boldsymbol{z}^m$. This equation is an even more abstract form of (2.122). Due to the non-linearities (concerning both the grid update and the non-linear part), it must be solved using a Picard-type iterative method. We consider the following update rule:

$$Q[\zeta^\ell]\zeta^{\ell+1} + q[\zeta^\ell](\zeta^\ell) = \widetilde{f}[\zeta^\ell], \tag{2.124}$$

where $\zeta^0 = \boldsymbol{z}^{m-1}$. Note that $\zeta^{\ell+1}$ can be obtained by solving a linear system. Using this method, we only need a single Picard-loop in each timestep. The algorithm is presented in pseudocode in Algorithm 1.

Algorithm 1 Single Loop Timestep

```
set ζ^0 = z^t
for ℓ = 0, 1, ... do
    ζ^{ℓ+1} = Q[ζ^ℓ]^{-1}(f̃[ζ^ℓ] − q[ζ^ℓ](ζ^ℓ))
    if ‖ζ^{ℓ+1} − ζ^ℓ‖ < δ then
        set z^{t+Δt} = ζ^{ℓ+1}
        break for-loop
    end if
end for
```

Another option would be to use a different update rule, such as

$$Q[\zeta^\ell]\zeta^{\ell+1} + q[\zeta^\ell](\zeta^{\ell+1}) = \widetilde{f}[\zeta^\ell]. \tag{2.125}$$

Now, in order to calculate $\zeta^{\ell+1}$, one must invert the non-linear operator $q[\zeta^\ell](\cdot)$. As this is not feasible in practice, a nested iterative scheme must be used to obtain $\zeta^{\ell+1}$. Such a nested iterative process is presented in Algorithm 2.

Algorithm 2 Nested Loop Timestep

```
set ζ^0 = z^t
for ℓ = 0, 1, ... do
    set ζ^{ℓ+1,0} = ζ^ℓ
    for m = 0, 1, ... do
        ζ^{ℓ+1,m+1} = Q[ζ^ℓ]^{-1}(f̃[ζ^ℓ] − q[ζ^ℓ](ζ^{ℓ+1,m}))
        if ‖ζ^{ℓ+1,m+1} − ζ^{ℓ+1,m}‖ < δ_1 then
            set ζ^{ℓ+1} = ζ^{ℓ+1,m+1}
            break for-loop
        end if
    end for
    if ‖ζ^{ℓ+1} − ζ^ℓ‖ < δ_2 then
        set z^{t+Δt} = ζ^{ℓ+1}
        break for-loop
    end if
end for
```

In practice we see that Algorithm 1 performs well. It should be noted that although an unconditionally stable method is used, we still have a restriction on the timestep. If Δt is too big, then the Picard iterations will not convergence.

2.4.4 Numerical Experiments

Let the initial domain be the unit square, $\Omega(0) = (0,1)^2$. The left border ($x = 0$) is fixed in place. A homogeneous Neumann boundary condition is imposed on the other boundaries; thus $\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{0}$. The body force $\boldsymbol{f}$ is set to

$$\boldsymbol{g}(\boldsymbol{x}, t) = \begin{pmatrix} F \\ 0 \end{pmatrix} \mathbb{1}_{[0,1]}(t),$$

where $F = 1/2$. We also set

$$\rho = 1, \quad \mu_\varepsilon = \lambda_\varepsilon = \mu_v = \lambda_v = 1.$$

Note that this corresponds to setting the Poisson ratio to $\nu = 1/3$, hence we expect some amount of transaxial compression as a consequence of axial stretching. We set $\boldsymbol{G} = \alpha\boldsymbol{\varepsilon}$ for various values of α. When $\alpha = 0$, we expect there to be no permanent deformations; the system is purely elastic. In this case, the domain should 'bounce back' to its original state after $t = 1$. For $\alpha > 0$, we expect there to be some permanent deformations. Figures 2.7, 2.8 and 2.9 show the finite element solution at $t = 10$ for $\alpha = 0$, $\alpha = 1$ and $\alpha = 10$ respectively.

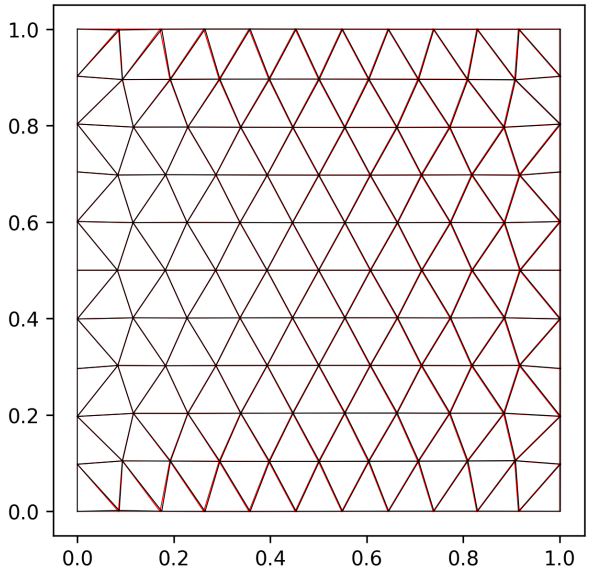


Figure 2.7: finite element solution at $t = 10$ for $\alpha = 0$.

Note that as expected, the domain has bounced back to its original state.

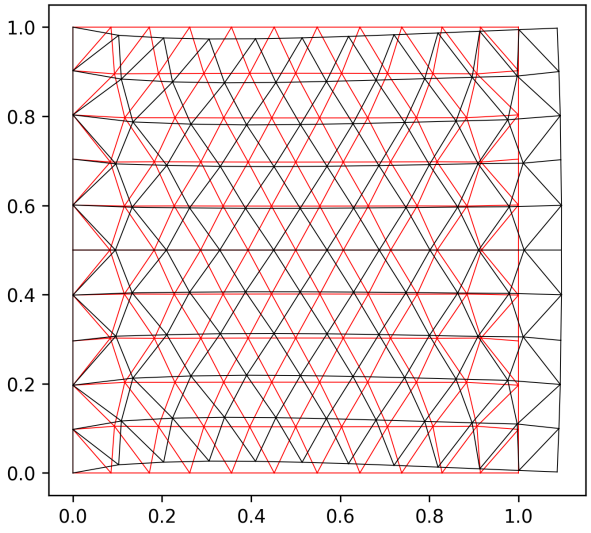


Figure 2.8: finite element solution at $t = 10$ for $\alpha = 1$.

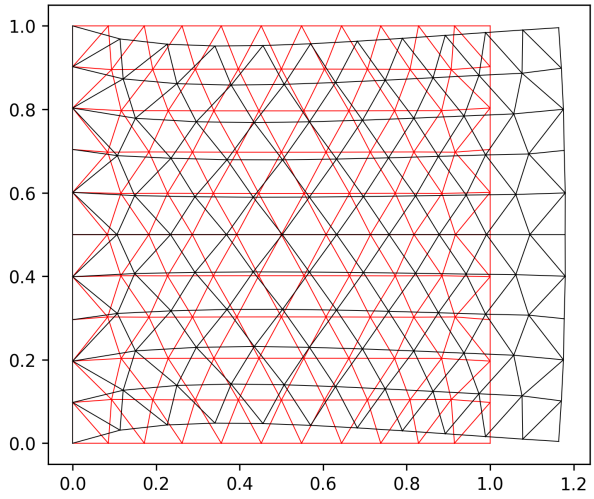


Figure 2.9: finite element solution at $t = 10$ for $\alpha = 10$.

2.5 Comparison Between Elastic Models

In this section, we show the qualitative differences between the three elastic models. To this end, a time-dependent force is applied to the domain. In the case of the purely elastic model, equation (2.14) is made time-dependent. That is, since the body force is a function of space and time; $\boldsymbol{g} = \boldsymbol{g}(\boldsymbol{x}, t)$, so is the displacement: $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, t)$. For the first comparison, we consider the unit square, and set the body force equal to

$$\boldsymbol{g}(\boldsymbol{x}, t) = \begin{pmatrix} 0 \\ -0.1 \sin(\pi t) \mathbb{1}_{[0,1]}(t) \end{pmatrix} \tag{2.126}$$

Thus for $t > 1$ no force is applied. The left boundary of the domain ($x = 0$) is fixed in place. The shear force on the rest of the boundary is set equal to zero. To compare the different models, we track the y-coordinate of the bottom right corner of the square. Moverover, a uniform initial grid with maxmimum meshwidth $h = 0.05\sqrt{2}$ is used. Figure 2.10 shows this grid, and the position of the tracked point.

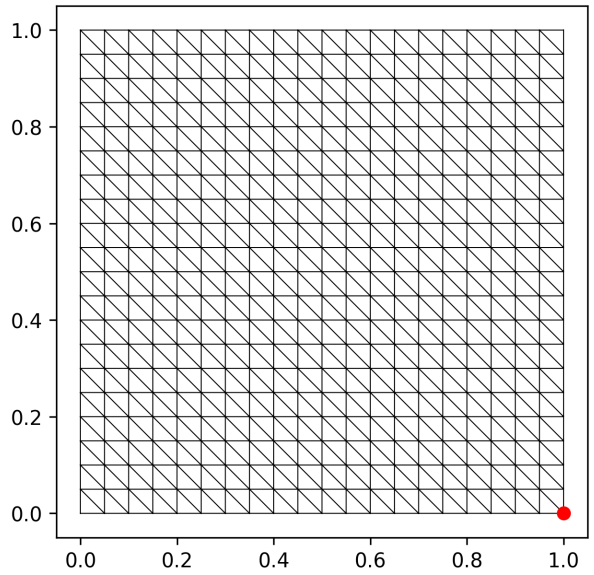


Figure 2.10: Finite element mesh with tracked grid node shown in red.

As the body force is applied to the domain, we expect it to move in a similar fashion to Figure 2.6. In the elastic case, the displacement will react instantaneously to the force, whereas in the visco- and morphoelastic models, the reaction will be delayed due to the interial forces. Furthermore, in the morphoelastic case we expect some permanent deformations, hence the domain will not return to its original shape. We

run the simulations until $t = 5$ using a timestep of size $\Delta t = 0.05$. The mechanical parameters are set to

$$\mu_\varepsilon = \lambda_\varepsilon = 1, \quad \mu_v = \lambda_v = 0.5, \quad \rho = 0.1. \tag{2.127}$$

Figure 2.11 shows the y-coordinate of the red point as a function of time. Note that we have considered two different morphoelastic models, one with $\alpha = 0.5$ and the other with $\alpha = 1$. We see that in the latter case, there is more permanent deformation.

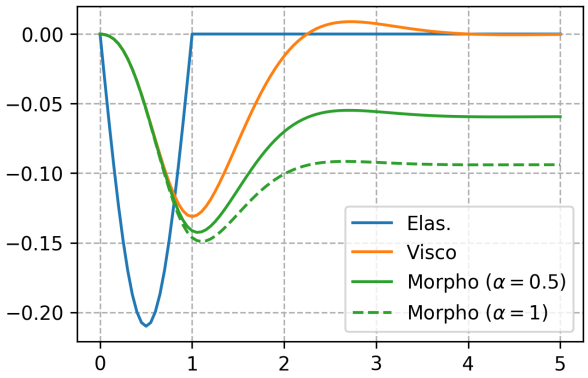


Figure 2.11: y-coordinate of bottom right corner as a function time for various elastic models.

Also observe that in the purely elastic model, the reaction the force is indeed instantaneous. After the body force vanishes, the domain will immediately return to its initial state. In the viscoelastic model, the domain also returns to the initial state, but this takes more time. Moreover, in both the viscoelastic and morphoelastic models, there is a small amount of 'overshoot'. Due to the inclusion of inertial forces, the domain slightly oscillates around the equilibrium position.

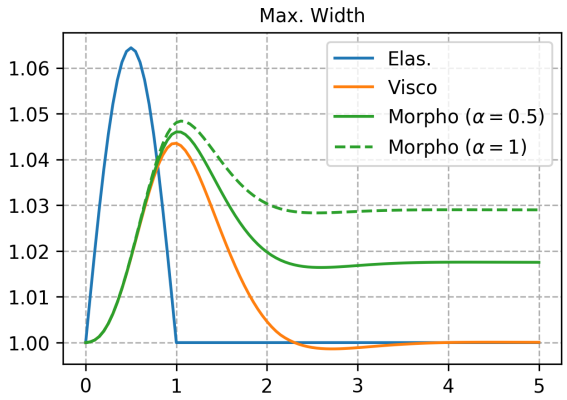
For the second comparison, we use the same initial domain and boundary condition. The body force is set to

$$\boldsymbol{g}(\boldsymbol{x}, t) = \begin{pmatrix} 0.2 \sin(\pi t) \mathbb{1}_{[0,1]}(t) \\ 0 \end{pmatrix}. \tag{2.128}$$

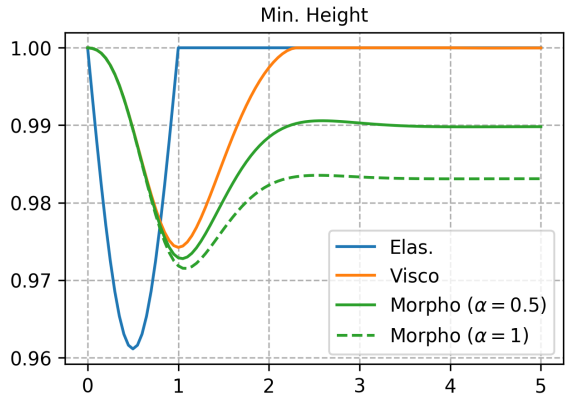
Thus, we now expect the domain to stretch in the x-direction. The following mechanical parameters are used:

$$\mu_\varepsilon = \lambda_\varepsilon = 1, \quad \mu_v = \lambda_v = 0.5, \quad \rho = 0.5. \tag{2.129}$$

Thus, due to λ_ε being positive, we also expect some transaxial compression. To measure the stretching and compressing of the domain, we track the maximum domain width (in the x-direction), and the minimum domain height (in the y-direction). Figures 2.12a and 2.12b show these two quantities as a function of time.



(a) Maximum width of domain.



(b) Minimum height of domain.

Figure 2.12: Maximum domain width and mimimum domain height as a function of time for various elastic models.

Note that in Figure 2.12b the line corresponding to the viscoelastic simulation contains a non-differentiable point. This is simply the result of fixing the left border. Hence there will always be a height of length 1 in the domain.

An interesting property of the morphoelastic model is that although the domain does not converge to its original state, and thus the displacement does not converge to zero, the strain energy *does* converge to zero. In Figure 2.13 the strain energy is shown for the morphoelastic simulation with $\alpha = 1$, thus it corresponds to the dashed green lines in Figures 2.12a and 2.12b. We define the elastic strain energy by

$$\begin{aligned} E_{\text{strain}}(t) &= \frac{1}{2} \int_{\Omega(t)} \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{\varepsilon}) : \boldsymbol{\varepsilon} \, \mathrm{d}\Omega \\ &= \frac{1}{2} \int_{\Omega(t)} \mu_\varepsilon \, \boldsymbol{\varepsilon} : \boldsymbol{\varepsilon} + \lambda_\varepsilon \mathrm{tr}(\boldsymbol{\varepsilon})^2 \, \mathrm{d}\Omega. \end{aligned} \tag{2.130}$$

This shows us that the classical definition of the strain tensor (2.8) does not make sense in a morphoelastic setting. The strain tensor is no longer directly related to the displacement.

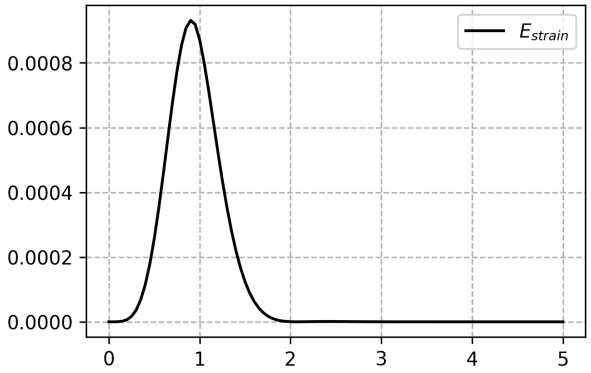


Figure 2.13: Strain energy as a function of time corresponding to the morphoelastic simulation with $\alpha = 1$.

3

The Mechanics of Porous Media

In this chapter a number of finite element models are developed for different poroelastic systems. We start with the one-dimensional Terzaghi problem, which is a simplified poroelastic model. It is solved using linear-linear elements, meaning that we use linear basis functions for both $\boldsymbol{u}$ and p, and quadratic-linear elements, where we use quadratic basis functions for $\boldsymbol{u}$ and linear for p. In both cases we encounter non-physical oscillations in the pressure solution for small values of k, or small timesteps. It is shown that these oscillations can be fixed by using diffusive stabilization.

We then move on to two-dimensional poroelasticity. First we consider infinitessimal deformations corresponding to high values of the Lamé constants. This poroelastic system is solved on a fixed domain, since the deformation tensor is assumed to be close to unity. We subsequently consider visco- poroelasticity on a moving domain. This model essentially combines poroelasticity with the viscoelastic model from Section 2.3. The chapter is concluded by a comparison between the viscoelastic and visco-poroelastic models.

3.1 Porous Media

In this section we will discuss the basic physical assumptions on which poroelastic systems are based. A porous medium is a material that contains small, connected voids. These voids, called pores, are completely filled with one or more fluids. Fluid flows through porous media are used in many different fields. For example in the field of geomechanics, the interaction between water and oil in a porous rock is often modeled using a two-phase fluid flow in a porous medium. In the context of this project the porous medium represents a tissue in which tumor cells are present. The poro-fluid is the extra-cellular fluid.

The pores in the medium are very small. We consider the porous medium from a macroscopic point of view, hence we are not interested in the precise locations and shapes of the pores. Instead we consider the parameter Φ, which represents the fraction of space occupied by solid material. Hence $1-\Phi$ is the fraction of space containing voids and is called the porosity. Instead of considering the velocity of individual fluid particles, an average poro-fluid flux called the Darcy velocity is used. It is denoted by $\boldsymbol{v}_D$. To relate the Darcy velocity to the actual poro-fluid velocity we need to take into account the porosity:

$$\boldsymbol{v}_F = \frac{\boldsymbol{v}_D}{1-\Phi}. \tag{3.1}$$

If Φ is close to 1 then there is not much space for the fluid to occupy. Thus to achieve a certain Darcy velocity, the fluid velocity must be high. Darcy's law relates the Darcy flux to the pressure gradient:

$$\boldsymbol{v}_D = -\frac{k_{\text{perm}}}{\mu_{\text{dyn}}}\nabla p. \tag{3.2}$$

Here k_{perm} is the permeability of the medium, and μ_{dyn} is the dynamic viscosity of the poro-fluid. Note that μ_{dyn} might be different from the viscoelastic parameter μ_v. The reason for this difference is that the medium might contain other fluids or materials, leading to a different viscosity of the medium compared to the poro-fluid. From this point, we will write $k = k_{\text{perm}}/\mu_{\text{dyn}}$. The parameter k is assumed to be constant in the remainder of this work. It will become important in Chapter 3: small values can lead to

problems with the solution. Darcy's law tells us that the Darcy velocity is proportional to the pressure gradient, and flows from high-pressure locations to low-pressure locations.

To relate $\boldsymbol{v}_D$ to the displacement velocity, we define the net velocity $\boldsymbol{v}_{\text{net}} = \boldsymbol{v} + \boldsymbol{v}_D$, where $\boldsymbol{v}$ represents the displacement velocity. The net velocity is assumed to be incompressible: $\nabla \cdot \boldsymbol{v}_{\text{net}} = 0$. This assumption follows from conservation of matter. By applying Darcy's law we find the equation

$$\nabla \cdot \boldsymbol{v} - \nabla \cdot (k \nabla p) = 0. \tag{3.3}$$

In the poroelastic systems from Chapter 3, we use the slightly adapted equation

$$\nabla \cdot \boldsymbol{v} - \nabla \cdot (k \nabla p) = f. \tag{3.4}$$

Here f represent a mass source (or sink) function. In a poroelastic medium the fluid pressure must also be taken into account in the force balance equation. For this reason the term $-p\boldsymbol{I}$ is added to the stress tensor.

3.2 Terzaghi Problem

We develop several different finite element models for the Terzaghi problem in one dimension. This problem is analyzed in [1] in order to show that non-physical oscillations occur for small timesteps. We consider the domain $\Omega = (0, 1)$, assume the medium is porous and purely elastic, and consider only infinitessimal deformations. The corresponding stress tensor in one dimension is given by:

$$\sigma = (\mu_\varepsilon + \lambda_\varepsilon) \frac{\partial u}{\partial x} - p. \tag{3.5}$$

The force balance equation then becomes

$$-\frac{\partial \sigma}{\partial x} = 0, \quad \text{hence} \quad -(\mu_\varepsilon + \lambda_\varepsilon) \frac{\partial u}{\partial x} + \frac{\partial p}{\partial x} = 0 \tag{3.6}$$

Using the fact that for infinitessimal deformations we have $v = \partial u / \partial t$, equation (3.3) becomes

$$\frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} \right) - k \frac{\partial^2 p}{\partial x^2} = 0. \tag{3.7}$$

The Terzaghi problem is obtained by setting $\mu_\varepsilon + \lambda_\varepsilon = 1$ and $k = 1$. We then get the following system of partial differential equations:

$$-\frac{\partial^2 u}{\partial x^2} + \frac{\partial p}{\partial x} = 0, \qquad x \in (0, 1),\ t > 0, \tag{3.8}$$

$$\frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} \right) - \frac{\partial^2 p}{\partial x^2} = 0, \qquad x \in (0, 1),\ t > 0. \tag{3.9}$$

The following boundary conditions are imposed:

$$\frac{\partial u}{\partial x}(0, t) = -1, \quad p(0, t) = 0, \tag{3.10}$$

$$u(1, t) = 0, \quad \frac{\partial p}{\partial x}(1, t) = 0, \tag{3.11}$$

as well as the initial condition $\partial u / \partial x(x, 0) = 0$. Define the following function spaces:

$$\mathcal{V}_0 = \{\varphi \in H^1(0, 1) : \varphi(1) = 0\}, \tag{3.12}$$

$$\mathcal{W}_0 = \{\psi \in H^1(0, 1) : \psi(0) = 0\}. \tag{3.13}$$

The weak form of the Terzaghi problem is given by:

For every $t > 0$ find $u(\cdot, t) \in \mathcal{V}_0$ and $p(\cdot, t) \in \mathcal{W}_0$ such that

$$\begin{aligned} \int_0^1 \frac{\partial u}{\partial x} \frac{\mathrm{d}\varphi}{\mathrm{d}x} + \frac{\partial p}{\partial x} \varphi \, \mathrm{d}x &= \varphi(0), \qquad \forall \varphi \in \mathcal{V}_0, \\ \int_0^1 \frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} \right) \psi + \frac{\partial p}{\partial x} \frac{\mathrm{d}\psi}{\mathrm{d}x} \, \mathrm{d}x &= 0, \qquad \forall \psi \in \mathcal{W}_0. \end{aligned} \tag{3.14}$$

Note that the $\varphi(0)$ in the right-hand side of the first equation is a result of boundary condition (3.10). Let u_h and p_h be the finite element solutions. We write

$$u_h(x,t) = \sum_{j=1}^{n_u} u_j(t)\varphi_j(x), \qquad p_h(x,t) = \sum_{j=1}^{n_p} p_j(t)\psi_j(x), \tag{3.15}$$

where the φ_j span a finite dimensional subspace of $\mathcal{V}_0$ and the ψ_j span a finite dimensional subspace of $\mathcal{W}_0$. The Galerkin equations corresponding to the weak form (3.14) are given by

$$\begin{aligned} \sum_{j=1}^{n_u} u_j \int_0^1 \frac{\mathrm{d}\varphi_i}{\mathrm{d}x}\frac{\mathrm{d}\varphi_j}{\mathrm{d}x}\,\mathrm{d}x + \sum_{j=1}^{n_p} p_j \int_0^1 \varphi_i \frac{\mathrm{d}\psi_j}{\mathrm{d}x}\,\mathrm{d}x = \varphi_i(0), \qquad & i \in \{1,\dots,n_u\}, \\ \sum_{j=1}^{n_u} \frac{\mathrm{d}u_j}{\mathrm{d}t} \int_0^1 \psi_i \frac{\mathrm{d}\varphi_j}{\mathrm{d}x}\,\mathrm{d}x + \sum_{j=1}^{n_p} p_j \int_0^1 \frac{\mathrm{d}\psi_i}{\mathrm{d}x}\frac{\mathrm{d}\psi_j}{\mathrm{d}x}\,\mathrm{d}x = 0, \qquad & i \in \{1,\dots,n_p\}. \end{aligned} \tag{3.16}$$

In block matrix form we have

$$\begin{pmatrix} 0 & 0 \\ D & 0 \end{pmatrix} \begin{pmatrix} \dot{\boldsymbol{u}} \\ \dot{\boldsymbol{p}} \end{pmatrix} + \begin{pmatrix} S & G \\ 0 & L \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \\ \boldsymbol{p} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{0} \end{pmatrix}. \tag{3.17}$$

Here $\boldsymbol{u}$ and $\boldsymbol{p}$ are the vectors

$$\boldsymbol{u} = (u_1(t), u_2(t), \dots, u_{n_u}(t))^\top, \tag{3.18}$$

$$\boldsymbol{p} = \big(p_1(t), p_2(t), \dots, p_{n_p}(t)\big)^\top. \tag{3.19}$$

The blocks are given component-wise by:

$$D_{ij} = \int_0^1 \psi_i \frac{\mathrm{d}\varphi_j}{\mathrm{d}x}\,\mathrm{d}x, \qquad i \in \{1,\dots,n_p\},\ j \in \{1,\dots,n_u\}, \tag{3.20}$$

$$G_{ij} = \int_0^1 \varphi_i \frac{\mathrm{d}\psi_j}{\mathrm{d}x}\,\mathrm{d}x, \qquad i \in \{1,\dots,n_u\},\ j \in \{1,\dots,n_p\}, \tag{3.21}$$

$$S_{ij} = \int_0^1 \frac{\mathrm{d}\varphi_i}{\mathrm{d}x}\frac{\mathrm{d}\varphi_j}{\mathrm{d}x}\,\mathrm{d}x, \qquad i,j \in \{1,\dots,n_u\}, \tag{3.22}$$

$$L_{ij} = \int_0^1 \frac{\mathrm{d}\psi_i}{\mathrm{d}x}\frac{\mathrm{d}\psi_j}{\mathrm{d}x}\,\mathrm{d}x, \qquad i,j \in \{1,\dots,n_p\}. \tag{3.23}$$

The right-hand side vector $\boldsymbol{b}$ is given component-wise by:

$$b_i = \varphi_i(0). \tag{3.24}$$

In practice we will use basis functions that equal 1 at exactly one grid point and zero everywhere else. In this case $\boldsymbol{b}$ will have a 1 at its first position, and contain zeroes everywhere else. From (3.18) and (3.19) it follows that D is a $n_p \times n_u$ matrix, G is $n_u \times n_p$, S is $n_u \times n_u$, and L is $n_p \times n_p$. We apply implicit Euler to the system in (3.17). This leads to the following system:

$$\begin{pmatrix} S & G \\ D & \Delta t L \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ D & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{0} \end{pmatrix}. \tag{3.25}$$

The superscript denotes a time level.

3.2.1 Linear-Linear Elements

We first use linear basis functions for u and p. In this we have $n_p = n_u$. Let $\{0 = x_1 < x_2 < \cdots < x_{n+1} = 1\}$ be a uniform partition of $[0,1]$ with meshwidth h. We denote the linear basis function centered in x_k by λ_k. Let e_k be the element $[x_k, x_{k+1}]$ for $k \in \{1,\dots,n\}$. On e_k, only the functions λ_k and λ_{k+1} are nonzero. Futhermore, on e_k we have

$$\frac{\mathrm{d}\lambda_k}{\mathrm{d}x} = -\frac{1}{h}, \quad \frac{\mathrm{d}\lambda_{k+1}}{\mathrm{d}x} = \frac{1}{h}. \tag{3.26}$$

The element matrices corresponding to (3.20)-(3.23) are given by:

$$D^{e_k} = G^{e_k} = \frac{1}{2}\begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, \tag{3.27}$$

$$S^{e_k} = L^{e_k} = \frac{1}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{3.28}$$

In [1] it is shown that in order to obtain an oscillation-free solution, we must have

$$\Delta t > \frac{h^2}{4}. \tag{3.29}$$

Figure 3.1 and 3.2 show the finite element solution after one timestep, using a meshwidth of $h = 1/10$ and a timestep of $\Delta t = 1/200$ and $\Delta t = 1/800$ respectively. Note that in Figure 3.1 the bound in (3.29) holds, whereas in Figure 3.2 it does not.

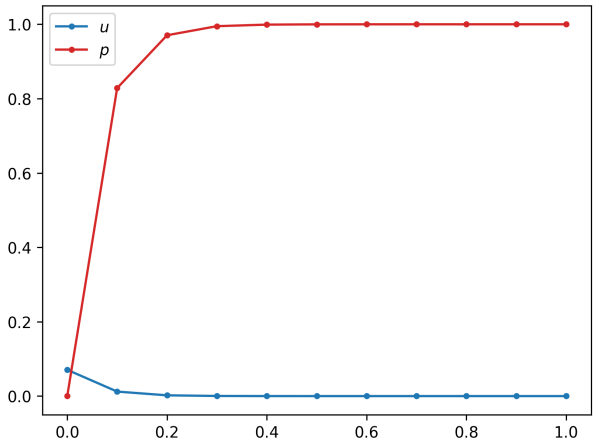


Figure 3.1: FE-solution after one timestep using $h = 1/10$ and $\Delta t = 1/200$.

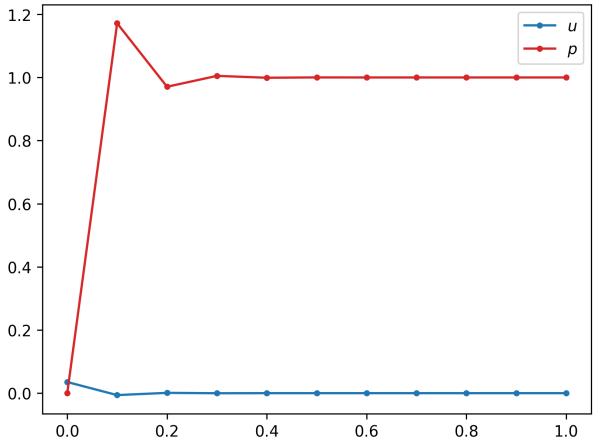


Figure 3.2: FE-solution after one timestep using $h = 1/10$ and $\Delta t = 1/800$.

3.2.2 Quadratic-Linear Elements

We now use quadratic basis functions for u and linear basis functions for p. Such elements are called Taylor-Hood elements. Since quadratic basis functions need an additions control vertex, we have $n_u = 2n_p - 1$. We again use a uniform grid with meshwidth h. Let $x_{k+\frac{1}{2}}$ be the midpoint of the element $e_k = [x_k, x_{k+1}]$. The quadratic basis functions $\varphi_k, \varphi_{k+\frac{1}{2}}, \varphi_{k+1}$ are defined by the fact that they equal 1 in their respective grid point and zero on the other two points. They can be expressed in terms of the

linear basis functions:

$$\varphi_k = \lambda_k(2\lambda_k - 1), \tag{3.30}$$
$$\varphi_{k+\frac{1}{2}} = 4\lambda_k\lambda_{k+1}, \tag{3.31}$$
$$\varphi_{k+1} = \lambda_{k+1}(2\lambda_{k+1} - 1). \tag{3.32}$$

Figure 3.3 shows the quadratic basis functions on the element e_k.

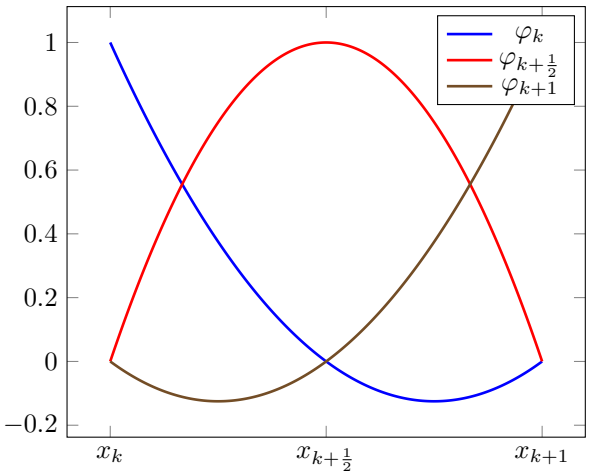


Figure 3.3: Quadratic basis functions on the element e_k.

From these expressions one can easily compute the derivatives of the quadratic basis functions:

$$\frac{\mathrm{d}\varphi_k}{\mathrm{d}x} = -\frac{1}{h}(4\lambda_k - 1), \tag{3.33}$$
$$\frac{\mathrm{d}\varphi_{k+\frac{1}{2}}}{\mathrm{d}x} = \frac{4}{h}(\lambda_k - \lambda_{k+1}), \tag{3.34}$$
$$\frac{\mathrm{d}\varphi_{k+1}}{\mathrm{d}x} = \frac{1}{h}(4\lambda_{k+1} - 1). \tag{3.35}$$

Now Holand & Bell's theorem [17] can be used to compute the element matrices. Note that most element matrices are no longer 2×2. For example, D^{e_k} is now of the form

$$D^{e_k} = \begin{bmatrix} D^{e_k}_{k,k} & D^{e_k}_{k,k+\frac{1}{2}} & D^{e_k}_{k,k+1} \\ D^{e_k}_{k+1,k} & D^{e_k}_{k+1,k+\frac{1}{2}} & D^{e_k}_{k+1,k+1} \end{bmatrix}, \tag{3.36}$$

since D is multiplied with $\boldsymbol{u}$, which now also contains coefficients corresponding to the midpoints of each element. The element matrices corresponding to (3.20)-(3.23) in the case of quadratic basis functions for u and linear for p are given by:

$$D^{e_k} = \frac{1}{6}\begin{bmatrix} -5 & 4 & 1 \\ -1 & -4 & 5 \end{bmatrix}, \tag{3.37}$$
$$G^{e_k} = \frac{1}{6}\begin{bmatrix} -1 & 1 \\ -4 & 4 \\ -1 & 1 \end{bmatrix}, \tag{3.38}$$
$$S^{e_k} = \frac{1}{3h}\begin{bmatrix} 7 & -8 & 1 \\ -8 & 16 & -8 \\ 1 & -8 & 7 \end{bmatrix}, \tag{3.39}$$
$$L^{e_k} = \frac{1}{h}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{3.40}$$

In [1] it is shown that the lower bound for Δt in the case of quadratic-linear elements is given by

$$\Delta t > \frac{h^2}{6}. \tag{3.41}$$

While this bound allows for smaller timesteps compared to the linear-linear case, it is only reduced by a factor 2/3. Figures 3.4 and 3.5 again show the finite element solutions after one timestep, using the same parameters as in Figures 3.1 and 3.2.

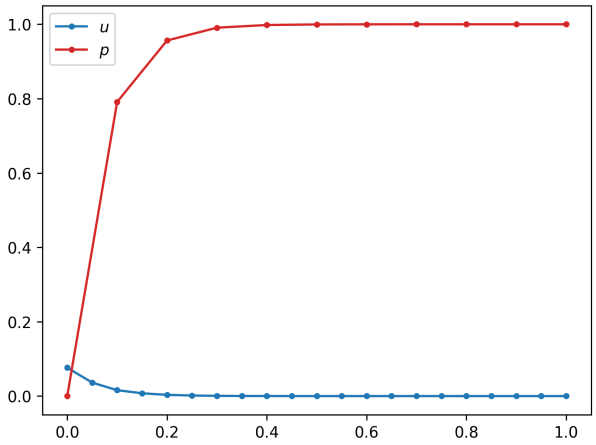


Figure 3.4: FE-solution after one timestep using $h = 1/10$ and $\Delta t = 1/200$.

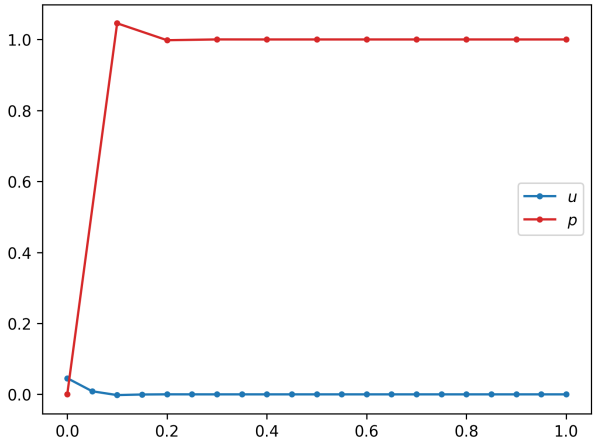


Figure 3.5: FE-solution after one timestep using $h = 1/10$ and $\Delta t = 1/800$.

Note that oscillations still occur for $\Delta t = 1/800$, but they are smaller in magnitude compared to the linear-linear case.

3.2.3 Stabilization

In [13] it is shown that the non-physical oscillations can be removed by introducing a stabilization term to equation (3.9). In the case of a uniform grid, equation (3.9) is replaced by:

$$\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial x}\right) - \frac{\partial^2 p}{\partial x^2} - \beta h^2 \frac{\partial}{\partial t}\left(\frac{\partial^2 p}{\partial x^2}\right) = 0, \tag{3.42}$$

where h is the meshwidth and β is a constant that depends on the spatial discretization. For linear-linear elements we have $\beta = 1/4$ and for Taylor-Hood elements we have $\beta = 1/6$. The corresponding Galerkin equations are then given by:

$$\begin{pmatrix} 0 & 0 \\ D & L_s \end{pmatrix} \begin{pmatrix} \dot{\boldsymbol{u}} \\ \dot{\boldsymbol{p}} \end{pmatrix} + \begin{pmatrix} S & G \\ 0 & L \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \\ \boldsymbol{p} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{0} \end{pmatrix}. \tag{3.43}$$

Notice that the difference with (3.17) is the L_s block in the first matrix. It is given by:

$$L_s = \beta h^2 L. \tag{3.44}$$

We note once again that these expressions are only valid in the case of a uniform grid. Applying backwards Euler to (3.43) yields

$$\begin{pmatrix} S & G \\ D & L_s + \Delta t L \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ D & L_s \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{0} \end{pmatrix}. \tag{3.45}$$

Figures 3.6 and 3.7 show the finite element solutions for linear-linear and quadratic-linear elements after one very small timestep. In both cases, the stabilized solutions show no oscillations.

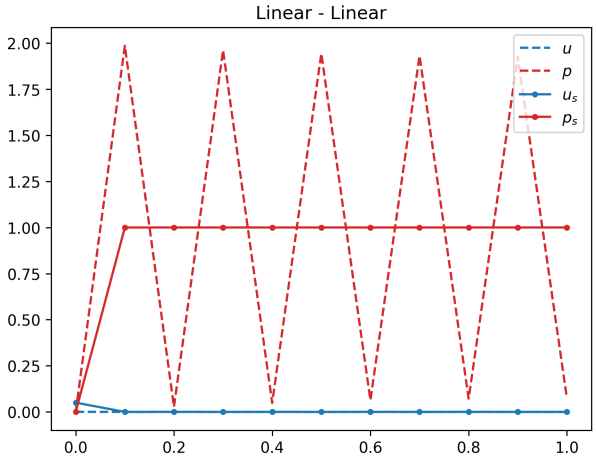


Figure 3.6: Stabilized and non-stabilized FE-solutions using linear-linear elements, after one timestep using $h = 1/10$ and $\Delta t = 10^{-6}$. The solid lines represent the stabilized solutions and the dashed lines represent the non-stabilized solutions

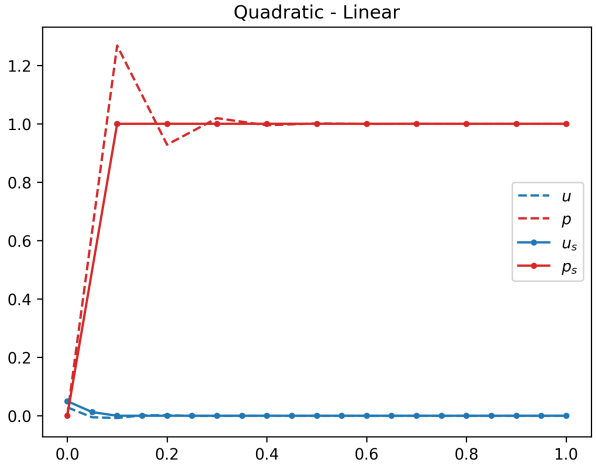


Figure 3.7: Stabilized and non-stabilized FE-solutions using quadratic-linear elements, after one timestep using $h = 1/10$ and $\Delta t = 10^{-6}$. The solid lines represent the stabilized solutions and the dashed lines represent the non-stabilized solutions

3.3 Poroelasticity on a Fixed Domain

Now consider a two-dimensional poroelastic system in the framework of infinitesimal deformations: it is assumed that $\|\nabla \boldsymbol{u}\| \ll 1$. Therefore the displacement velocity simplifies to

$$\boldsymbol{v} \approx \frac{\partial \boldsymbol{u}}{\partial t}, \tag{3.46}$$

by the derivation in (2.46). For this approximation to be accurate, high values of the Lamé constants μ_ε and λ_ε are used. Combining the purely elastic model from Section (2.2) with the simplified version of

equation (3.4) yields:

$$-\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}, p) = \boldsymbol{g}, \qquad \text{on } \Omega, \tag{3.47}$$

$$\nabla \cdot \frac{\partial \boldsymbol{u}}{\partial t} - \nabla \cdot (k \nabla p) = f, \qquad \text{on } \Omega. \tag{3.48}$$

The stress tensor is given by:

$$\boldsymbol{\sigma}(\boldsymbol{u}, p) = \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{u}) - p\boldsymbol{I}, \tag{3.49}$$

where the elasticity tensor $\boldsymbol{\sigma}_{\text{el}}$ is given in (2.10). We impose the following boundary conditions:

$$\boldsymbol{u} = \boldsymbol{0}, \quad k \nabla p \cdot \boldsymbol{n} = 0, \qquad \text{on } \Gamma_1, \tag{3.50}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\tau}, \quad p = 0, \qquad \text{on } \Gamma_2, \tag{3.51}$$

We use Implicit Euler to discretize in time:

$$-\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}^m, p^m) = \boldsymbol{g}^m, \tag{3.52}$$

$$\nabla \cdot \boldsymbol{u}^m - \Delta t \nabla \cdot (k \nabla p^m) = \Delta t f^m + \nabla \cdot \boldsymbol{u}^{m-1}. \tag{3.53}$$

Here Δt denotes the timestep. Using the same notation as in [13] the weak form is given by:

For each $m \geqslant 1$ find $\boldsymbol{u}^m \in \left[H^1_{\Gamma_1}(\Omega)\right]^2$ and $p^m \in H^1_{\Gamma_2}(\Omega)$ such that

$$a(\boldsymbol{u}^m, \boldsymbol{v}) - (p^m, \nabla \cdot \boldsymbol{v}) = (\boldsymbol{g}^m, \boldsymbol{v}) + (\boldsymbol{\tau}^m, \boldsymbol{v})_{\Gamma_2}, \qquad \forall \boldsymbol{v} \in \left[H^1_{\Gamma_1}(\Omega)\right]^2, \tag{3.54}$$

$$(\nabla \cdot \boldsymbol{u}^m, q) + \Delta t b(p^m, q) = (\Delta t f^m + \nabla \cdot \boldsymbol{u}^{m-1}, q), \qquad \forall q \in H^1_{\Gamma_2}(\Omega). \tag{3.55}$$

The function spaces $H^1_{\Gamma_1}(\Omega)$ and $H^1_{\Gamma_2}(\Omega)$ are the Sobolev spaces of degree 1 with functions that vanish on Γ_1 and Γ_2 respectively. Moreover, the bilinear forms $a(\cdot, \cdot)$ and $b(\cdot, \cdot)$ are given by:

$$a(\boldsymbol{u}, \boldsymbol{v}) = \mu_\varepsilon \int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) \, \mathrm{d}\Omega + \lambda_\varepsilon \int_\Omega (\nabla \cdot \boldsymbol{u})(\nabla \cdot \boldsymbol{v}) \, \mathrm{d}\Omega, \tag{3.56}$$

$$b(p, q) = \int_\Omega k \nabla p \cdot \nabla q \, \mathrm{d}\Omega. \tag{3.57}$$

Finally, $(\cdot, \cdot)$ denotes the standard inner product on its respective Sobolev space, and

$$(\boldsymbol{\tau}, \boldsymbol{v})_{\Gamma_2} = \int_{\Gamma_2} \boldsymbol{\tau} \cdot \boldsymbol{v} \, \mathrm{d}\Gamma. \tag{3.58}$$

To obtain a finite element formulation, the infinite dimensional Sobolev spaces $\left[H^1_{\Gamma_1}(\Omega)\right]^2$ and $H^1_{\Gamma_2}(\Omega)$ are replaced by the finite dimensional ones V_h and Q_h respectively. We get the following formulation:

For each $m \geqslant 1$ find $\boldsymbol{u}_h^m \in V_h$ and $p_h^m \in Q_h$ such that

$$a(\boldsymbol{u}_h^m, \boldsymbol{v}) - (p_h^m, \nabla \cdot \boldsymbol{v}) = (\boldsymbol{g}^m, \boldsymbol{v}) + (\boldsymbol{\tau}^m, \boldsymbol{v})_{\Gamma_2}, \qquad \forall \boldsymbol{v} \in V_h, \tag{3.59}$$

$$(\nabla \cdot \boldsymbol{u}_h^m, q) + \Delta t b(p_h^m, q) = (\Delta t f^m + \nabla \cdot \boldsymbol{u}_h^{m-1}, q), \qquad \forall q \in Q_h. \tag{3.60}$$

In general, the domain Ω will be replaced by an approximate domain Ω_h. All integrations in (3.59) and (3.60) will be done over Ω_h. We derive the Galerkin equations for general basis functions φ_i and ψ_i for $\boldsymbol{u}$ and p respectively. Let n_u, n_p be the number of grid nodes for the displacement and pressure respectively. We define the finite element solutions by

$$(u^1)_h^m(\boldsymbol{x}) = \sum_{j=1}^{n_u} (u_j^1)^m \varphi_j(\boldsymbol{x}), \quad (u^2)_h^m(\boldsymbol{x}) = \sum_{j=1}^{n_u} (u_j^2)^m \varphi_j(\boldsymbol{x}), \quad p_h^m(\boldsymbol{x}) = \sum_{j=1}^{n_p} p_j^m \psi_j(\boldsymbol{x}). \tag{3.61}$$

The basis functions φ_j and ψ_j form a basis of V_h and Q_h respectively, such that they equal 1 in their corresponding grid node $\boldsymbol{x}_j$. To obtain the Galerkin equations, substitute the finite element solutions

into (3.54) and (3.55). Furthermore set $\boldsymbol{v} = (\varphi_i, 0)^\top$ and $\boldsymbol{v} = (0, \varphi_i)^\top$ for $i = 1, \ldots, n_u$ in (3.54) and set $q = \psi_i$ for $i = 1, \ldots, n_p$ in (3.55). We then obtain a linear system of the form

$$\begin{pmatrix} S_{\text{el}} & -D^\top \\ D & k\Delta t L \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} \boldsymbol{g}^m + \boldsymbol{\tau}^m \\ \Delta t \boldsymbol{f}^m \end{pmatrix} + \begin{pmatrix} 0 \\ D\boldsymbol{u}^{m-1} \end{pmatrix}. \tag{3.62}$$

Where $\boldsymbol{u}^m = \left((\boldsymbol{u}^1)^m, (\boldsymbol{u}^2)^m\right)^\top$ and

$$\boldsymbol{u}_1^m = \left((u_1^1)^m, (u_2^1)^m, \ldots, (u_{n_u}^1)^m\right)^\top, \tag{3.63}$$

$$\boldsymbol{u}_2^m = \left((u_1^2)^m, (u_2^2)^m, \ldots, (u_{n_u}^2)^m\right)^\top, \tag{3.64}$$

$$\boldsymbol{p}^m = \left(p_1^m, p_2^m, \ldots, p_{n_p}^m\right)^\top. \tag{3.65}$$

Moreover, the matrices S and D are the block matrices

$$S_{\text{el}} = \begin{pmatrix} S_{\text{el}}^{11} & S_{\text{el}}^{12} \\ S_{\text{el}}^{21} & S_{\text{el}}^{22} \end{pmatrix}, \qquad D = \begin{pmatrix} D^1 & D^2 \end{pmatrix}, \tag{3.66}$$

where the blocks $S_{\text{el}}^{11}, S_{\text{el}}^{12}, S_{\text{el}}^{21}$ and S_{el}^{22} are given element-wise by (2.30)-(2.33). We also write $\boldsymbol{g} = \left(\boldsymbol{g}^1, \boldsymbol{g}^2\right)^\top$ and $\boldsymbol{\tau} = \left(\boldsymbol{\tau}^1, \boldsymbol{\tau}^2\right)^\top$. The remaining blocks and vectors are given element-wise by:

$$(D^1)_{ij} = \int_\Omega \psi_i \frac{\partial \varphi_j}{\partial x} \, \mathrm{d}\Omega, \quad (D^2)_{ij} = \int_\Omega \psi_i \frac{\partial \varphi_j}{\partial y} \, \mathrm{d}\Omega, \quad i \in \{1, \ldots, n_p\},\ j \in \{1, \ldots, n_u\}. \tag{3.67}$$

$$(\boldsymbol{g}^1)_i = \int_\Omega g^1 \varphi_i \, \mathrm{d}\Omega, \qquad (\boldsymbol{g}^2)_i = \int_\Omega g^2 \varphi_i \, \mathrm{d}\Omega, \qquad i \in \{1, \ldots, n_u\}. \tag{3.68}$$

$$(\boldsymbol{\tau}^1)_i = \int_{\Gamma_2} \tau^1 \varphi_i \, \mathrm{d}\Gamma, \qquad (\boldsymbol{\tau}^2)_i = \int_{\Gamma_2} \tau^2 \varphi_i \, \mathrm{d}\Gamma, \qquad i \in \{1, \ldots, n_u\}. \tag{3.69}$$

$$(\boldsymbol{f})_i = \int_\Omega f \psi_i \, \mathrm{d}\Omega, \qquad i \in \{1, \ldots, n_p\}. \tag{3.70}$$

The matrix L is a Laplace matrix:

$$L_{ij} = \int_\Omega \nabla \psi_i \cdot \nabla \psi_j \, \mathrm{d}\Omega, \quad i, j \in \{1, \ldots, n_p\}. \tag{3.71}$$

3.3.1 Linear-Linear Elements

We investigate the monotonicity and stability of the finite element solution for linear-linear elements. To this end we introduce some helpful notations:

Definition 1. A *triangulation* $\mathcal{T}_h$ of Ω is a set of triangles such that:

$$\bigcup_{T \in \mathcal{T}_h} T =: \Omega_h \approx \Omega.$$

The parameter h is called the meshwidth, which is often defined as

$$h = \max_{T \in \mathcal{T}_h} \{\operatorname{diam}(T)\}.$$

Given a triangulation $\mathcal{T}_h$, we define the following polynomial function spaces on Ω_h:

Definition 2. The space of locally polynomial functions of degree k with respect to the triangulation $\mathcal{T}_h$ is defined by:

$$\mathbb{P}^k(\mathcal{T}_h) = \{\varphi \in H^1(\Omega_h) : \forall T \in \mathcal{T}_h : \varphi|_T \text{ is a polynomial of degree } k\}. \tag{3.72}$$

Set $V_h = \left[\mathbb{P}^1_{\Gamma_1}(\mathcal{T}_h)\right]^2$ and $Q_h = \mathbb{P}^1_{\Gamma_2}(\mathcal{T}_h)$, where the subscripts denote part of the boundary on which the functions vanish. Note that in this case we have $n_u = n_p$. In [1], the poroelastic problem (3.47)-(3.48) with boundary conditions (3.50)-(3.51) is solved numerically on $\Omega = (-4, 4)^2$, using the right-hand side

functions $\boldsymbol{g} = (0, -1)^\top$ and $f = 0$. The stress vector $\boldsymbol{\tau}$ on Γ_2 is set to $\mathbf{0}$. Furthermore, the boundary components are given by:

$$\Gamma_2 = \{(x, y) \in \Gamma : x = 4 \text{ or } y = 4\}, \qquad \Gamma_1 = \Gamma \setminus \Gamma_2.$$

We set $\mu_\varepsilon = \lambda_\varepsilon = 1000$ and vary the parameter k. In figures 3.8 and 3.9 the finite element solutions are shown after one timestep using a meshwidth of $h = 0.2$, for $k = 10^{-2}$ and $k = 10^{-8}$ respectively.

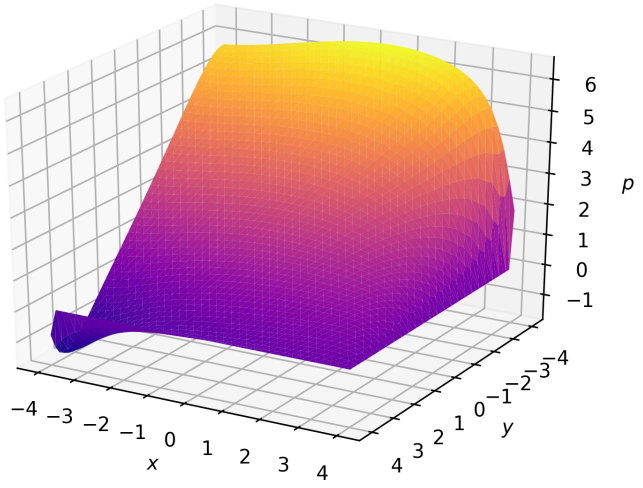


Figure 3.8: 3D-profile of the pressure at $t = 0.01$ for $k = 10^{-2}$.

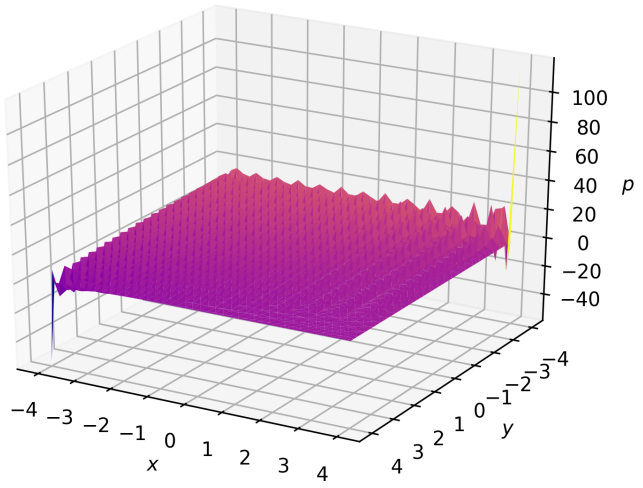


Figure 3.9: 3D-profile of the pressure at $t = 0.01$ for $k = 10^{-8}$.

We observe non-physical oscillations for $k = 10^{-8}$. These oscillations also occur for higher values of k if the timestep is reduced. These results are similar to the 1D case, where non-physical oscillations occur due the left-hand side matrix is no longer being an M-matrix for small timesteps. This phenomenon is a corollary of the fact that no inf-sup condition is satisfied when using linear-linear elements. In [13] an inf-sup condition is proved for the MINI element. This means that for this particular finite element discretization, no non-physical oscillations will occur: the solution is stabilized. In the next section, we will derive this discretization.

3.3.2 Stabilization with MINI Element

A MINI element is a triangular element with four nodes: its vertices and its barycenter. A generic MINI element is shown in Figure 3.10.

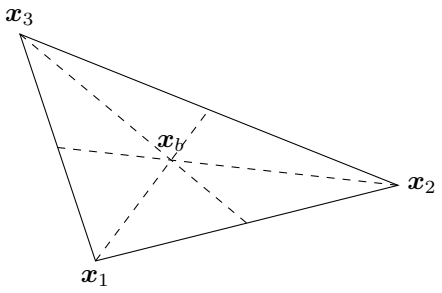


Figure 3.10: Nodes of a MINI element, where $\boldsymbol{x}_b$ is the barycenter of T

On this element, there are four basis functions; the linear barycentric coordinate functions that correspond to a vertex, and a qubic 'bubble function' that corresponds to the barycenter.

Definition 3. Let $T \in \mathcal{T}_h$ be a triangular element. Let $\lambda_1, \lambda_2, \lambda_3$ denote the barycentric coordinates of T, viewed as functions of position $\boldsymbol{x} \in T$. Then the *bubble function* $\varphi_{b,T}$ is defined as:

$$\varphi_{b,T} = 27\lambda_1\lambda_2\lambda_3. \tag{3.73}$$

The space of bubble functions w.r.t the triangulation $\mathcal{T}_h$ is given by

$$\mathbb{B}(\mathcal{T}_h) = \operatorname{Span}\left\{\varphi_{b,T} : T \in \mathcal{T}_h\right\}. \tag{3.74}$$

The function $\varphi_{b,T}$ is called a bubble function because it vanishes on the edges of T and it is equal to 1 at the barycenter of T. We use a superscripts to denote function spaces of vector-valued functions. For example, when we write $\boldsymbol{u} \in \left[\mathbb{P}^k(\mathcal{T}_h)\right]^2$, then in fact we mean $\boldsymbol{u} = (u^1, u^2)$ with $u^1, u^2 \in \mathbb{P}^k(\mathcal{T}_h)$.

Theorem 4. *Let $a(\cdot,\cdot)$ be bilinear form as defined in (3.56). Furthermore, let $\boldsymbol{u}_\ell, \boldsymbol{v}_\ell \in \left[\mathbb{P}^1(\mathcal{T}_h)\right]^2$ and $\boldsymbol{u}_b, \boldsymbol{v}_b \in \left[\mathbb{B}(\mathcal{T}_h)\right]^2$, then*

$$a(\boldsymbol{u}_\ell + \boldsymbol{u}_b, \boldsymbol{v}_\ell + \boldsymbol{v}_b) = a(\boldsymbol{u}_\ell, \boldsymbol{v}_\ell) + a(\boldsymbol{u}_b, \boldsymbol{v}_b).$$

Proof. Note that it suffices to show that $a(\boldsymbol{u}_\ell, \boldsymbol{v}_b) = 0$ and $a(\boldsymbol{u}_b, \boldsymbol{v}_\ell) = 0$. Since $a(\cdot,\cdot)$ is symmetric, only one of these equalities has to be proved. From its definition, it is clear that $a(\boldsymbol{u}_\ell, \boldsymbol{v}_b)$ is a weighted sum of integrals of the form

$$\int_{\Omega_h} \frac{\partial u_\ell^i}{\partial x_\alpha} \frac{\partial v_b^j}{\partial x_\beta} \,\mathrm{d}\Omega,$$

where $i, j, \alpha, \beta \in \{1, 2\}$. We split this integral into integrals over every triangle $T \in \mathcal{T}_h$:

$$\int_{\Omega_h} \frac{\partial u_\ell^i}{\partial x_\alpha} \frac{\partial v_b^j}{\partial x_\beta} \,\mathrm{d}\Omega = \sum_{T \in \mathcal{T}_h} \int_T \frac{\partial u_\ell^i}{\partial x_\alpha} \frac{\partial v_b^j}{\partial x_\beta} \,\mathrm{d}\Omega.$$

We will use the integration by parts formula combined with Gauss' divergence theorem:

$$\int_T \xi(\nabla \cdot \boldsymbol{F}) \,\mathrm{d}\Omega = \int_{\partial T} \xi(\boldsymbol{n} \cdot \boldsymbol{F}) \,\mathrm{d}\Gamma - \int_T \nabla\xi \cdot \boldsymbol{F} \,\mathrm{d}\Omega.$$

For $\beta = 1$ let $\boldsymbol{F} = (v_b^j, 0)$, and for $\beta = 2$ let $\boldsymbol{F} = (0, v_b^j)$. Moreover, let $\xi = \partial u_\ell^i / \partial x_\alpha$. Then by the above formula we get

$$\int_T \frac{\partial u_\ell^i}{\partial x_\alpha} \frac{\partial v_b^j}{\partial x_\beta} \,\mathrm{d}\Omega = \int_{\partial T} \frac{\partial v_\ell^i}{\partial x_\alpha} n_\beta v_b^j \,\mathrm{d}\Gamma - \int_T \frac{\partial^2 u_\ell^i}{\partial x_\beta \partial x_\alpha} v_b^j \,\mathrm{d}\Omega.$$

Since v_b^j is a bubble function it vanishes on the boundary ∂T. Thus the first integral equals zero. Furthermore, since u_ℓ^i is linear on T all its second derivatives vanish. Therefore the second integral also equals zero. We conclude that on each triangle T the above integral vanishes, and thus $a(\boldsymbol{u}_\ell, \boldsymbol{v}_b) = 0$. □

To stabilize the solution we set $V_h = \left[\mathbb{P}^1(\mathcal{T}_h) \oplus \mathbb{B}(\mathcal{T}_h)\right]^2$ and $Q_h = \mathbb{P}^1(\mathcal{T}_h)$. By Theorem 4 we obtain a system of the form:

$$\begin{pmatrix} S_b & 0 & -D_b^\top \\ 0 & S_{\mathrm{el}} & -D^\top \\ D_b & D & k\Delta t L \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_b^m \\ \boldsymbol{u}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ D_b & D & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_b^{m-1} \\ \boldsymbol{u}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \boldsymbol{g}_b^m \\ \boldsymbol{g}^m + \boldsymbol{\tau}^m \\ \Delta t \boldsymbol{f}^m \end{pmatrix} \tag{3.75}$$

Note that there is no vector $\boldsymbol{\tau}_b$ because it would involve boundary integrals of bubble functions, which vanish. Also note that there are just as many 'bubble-unknowns' as there are triangles in the triangulation. We denote this number by n_T. Let n denote the number of $\mathbb{P}^1$-unknowns, which corresponds to the number of nodes in the triangulation. Then S_b is a $2n_T \times 2n_T$ matrix that can be subdivided into four blocks like in (3.66). Moreover D_b is a $n \times 2n_T$ matrix. The per-element contributions of the newly introduced bubble matrices and vectors can be found in Section A.3.

Elimination of Bubbles
Instead of solving (3.75), one can eliminate the bubble equations to obtain a smaller system.

Theorem 5. *If* $(\boldsymbol{u}_b^m, \boldsymbol{u}^m, \boldsymbol{p}^m)^\top$ *solves* (3.75)*, then* $(\boldsymbol{u}^m, \boldsymbol{p}^m)^\top$ *solves:*

$$\begin{pmatrix} S_{el} & -D^\top \\ D & k\Delta t L + C_b \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ D & C_b \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \boldsymbol{g}^m + \boldsymbol{\tau}^m \\ \Delta t \boldsymbol{f}^m - B_b(\boldsymbol{g}_b^m - \boldsymbol{g}_b^{m-1}) \end{pmatrix}, \tag{3.76}$$

where

$$C_b = D_b S_b^{-1} D_b^\top, \qquad B_b = D_b S_b^{-1}. \tag{3.77}$$

Proof. Note that we can rewrite the first block-equation of (3.75) to:

$$S_b \boldsymbol{u}_b^m = D_b^\top \boldsymbol{p}^m + \boldsymbol{g}_b^m.$$

And thus

$$\begin{aligned} D_b \boldsymbol{u}_b^m = D_b S_b^{-1} S_b \boldsymbol{u}_b^m &= D_b S_b^{-1} D_b^\top \boldsymbol{p}^m + D_b S_b^{-1} \boldsymbol{g}^m \\ &= C_b \boldsymbol{p}^m + B_b \boldsymbol{g}_b^m. \end{aligned}$$

We will later give an explicit expression for S_b^{-1}. Substituting the above expression into the third block-equation of (3.75) yields:

$$D\boldsymbol{u}^m + (k\Delta t L + C_b)\boldsymbol{p}^m = D\boldsymbol{u}^{m-1} + C_b \boldsymbol{p}^{m-1} + \Delta t \boldsymbol{f}^m - B_b(\boldsymbol{g}_b^m - \boldsymbol{g}_b^{m-1}).$$

Thus, we have eliminated the first block-equation and obtain the system in (3.76). □

We give an explicit expression for S_b^{-1}. We have already seen that S_b consists of four diagonal blocks. From (A.37) and (A.38) it is evident that $S_b^{12} = S_b^{21}$. In other words S_b is symmetric. We have the following Lemma:

Lemma 4. *Let* S_b *be the matrix as defined by* (A.36)-(A.39)*, and let*

$$\Lambda = S_b^{11} S_b^{22} - (S_b^{12})^2. \tag{3.78}$$

If Λ *is invertible, then*

$$S_b^{-1} = \begin{pmatrix} \Lambda^{-1} & 0 \\ 0 & \Lambda^{-1} \end{pmatrix} \begin{pmatrix} S_b^{22} & -S_b^{12} \\ -S_b^{12} & S_b^{11} \end{pmatrix} \tag{3.79}$$

Proof. Note that since all blocks are diagonal and have the same dimensions, they all commute.

$$\begin{aligned} S^{-1}S &= \begin{pmatrix} \Lambda^{-1} & 0 \\ 0 & \Lambda^{-1} \end{pmatrix} \begin{pmatrix} S_b^{22} & -S_b^{12} \\ -S_b^{12} & S_b^{11} \end{pmatrix} \begin{pmatrix} S_b^{11} & S_b^{12} \\ S_b^{12} & S_b^{22} \end{pmatrix} \\ &= \begin{pmatrix} \Lambda^{-1} & 0 \\ 0 & \Lambda^{-1} \end{pmatrix} \begin{pmatrix} S_b^{11} S_b^{22} - (S_b^{12})^2 & 0 \\ 0 & S_b^{11} S_b^{22} - (S_b^{12})^2 \end{pmatrix} \\ &= \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} \end{aligned}$$

□

Note that we need Λ to be invertible. In practice we find that this is always the case, but we have yet to prove this fact.

3.3.3 Diffusive Stabilization

A different method used to stabilize the system is to introduce a small diffusive pertubation to the flow equation, similar to the one-dimensional case. In [1] it is shown that on a uniform grid the perturbed flow equation is given by:

$$\nabla \cdot \frac{\partial \boldsymbol{u}}{\partial t} - \nabla \cdot (k \nabla p) = f + \beta h^2 \frac{\partial}{\partial t} \left(\nabla^2 p \right) \tag{3.80}$$

Note that β is a tuning parameter. In [1] it is shown that in the one-dimensional case the optimal value of β is given by

$$\beta = \frac{1}{4(\lambda_\varepsilon + \mu_\varepsilon)}. \tag{3.81}$$

Numerical experiments show that this value is also optimal in 2D on a uniform grid [1, Figure 5]. Note that an error of order h^2 is introduced as a result of the added pertubation in (3.80). Since the finite element error is already of order h^2 when using linear-linear elements, the effect of the pertubation will be minimal. Discretization on a uniform grid using linear-linear elements and applying implicit Euler yields a system of the form

$$\begin{pmatrix} S & -D^\top \\ D & (k\Delta t + \beta h^2) L \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ D & \beta h^2 L \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \boldsymbol{g}^m + \boldsymbol{\tau}^m \\ \Delta t \boldsymbol{f}^m \end{pmatrix}. \tag{3.82}$$

Here L is a Laplace matrix:

$$L_{ij} = \int_{\Omega_h} \nabla \psi_i \cdot \nabla \psi_j \, \mathrm{d}\Omega. \tag{3.83}$$

The stabilization term $\beta h^2 L$ can be generalized to non-uniform grids by replacing it with the matrix C_s, given by:

$$(C_s)_{ij} = \beta \sum_{T \in \mathcal{T}} h_T^2 \int_T \nabla \psi_i \cdot \nabla \psi_j \, \mathrm{d}\Omega. \tag{3.84}$$

Here $\mathcal{T}$ is a (non-uniform) triangulation of Ω, and h_T denotes the diameter of a triangle T. System (3.82) is then given by

$$\begin{pmatrix} S & -D^\top \\ D & k\Delta t L + C_s \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ D & C_s \end{pmatrix} \begin{pmatrix} \boldsymbol{u}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \boldsymbol{g}^m + \boldsymbol{\tau}^m \\ \Delta t \boldsymbol{f}^m \end{pmatrix}. \tag{3.85}$$

Note that if $\mathcal{T}$ is uniform, we again obtain $C_s = \beta h^2 L$.

3.3.4 Comparison Between Bubble- and Diffusive Stabilization

We test both bubble stabilization (3.76) and diffusive stabilization (3.85) and compare the corresponding finite element solutions for p to the case where no stabilization is applied. To this end we solve equations (3.47)-(3.48) on $\Omega = (0,1)^2$. We set $\boldsymbol{g} = (0,-1)^\top$ and $f = 0$. Concerning the boundary conditions (3.50)-(3.51), we let $\Gamma_2 = \{(x,y) : x = 0, y \in [0,1]\}$ and $\Gamma_1 = \partial\Omega \setminus \Gamma_2$. The stress vector $\boldsymbol{\tau}$ on Γ_2 is set to $\mathbf{0}$. The parameter k is made small to cause oscillations in the non-stabilized solution: we set $k = 10^{-8}$. To ensure only small deformations occur, the Lamé constants are set to $\mu_\varepsilon = \lambda_\varepsilon = 1000$. A uniform triangulation is used with 21 grid nodes in each coordinate direction. Thus the meshwidth is equal to $h = 0.05\sqrt{2}$. A timestep of $\Delta t = 10^{-2}$ is used.

Figures 3.11, 3.12 and 3.13 show the finite element solutions for p using respectively no stabilization, bubble stabilization and diffusive stabilization. That is, in Figure 3.11 the system in (3.62) is solved, in Figure 3.12 the system in (3.76) is solved, and in Figure 3.13 the system in (3.85) is solved.

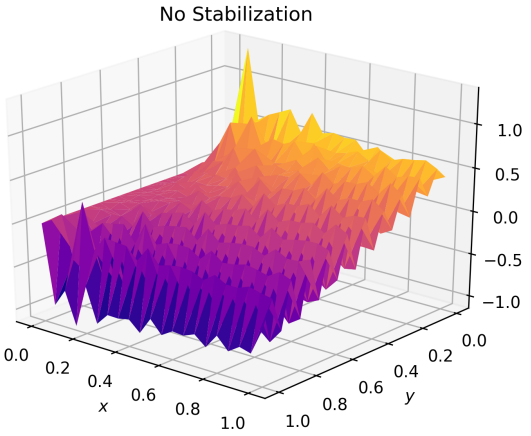


Figure 3.11: Finite element solution for p after one timestep of size $\Delta t = 10^{-2}$, using linear-linear elements.

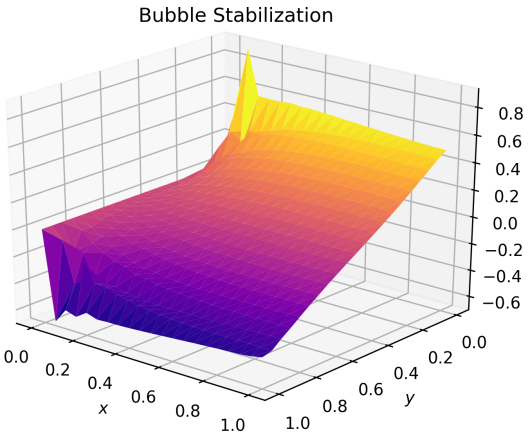


Figure 3.12: Finite element solution for p after one timestep of size $\Delta t = 10^{-2}$, using linear-linear elements and bubble stabilization.

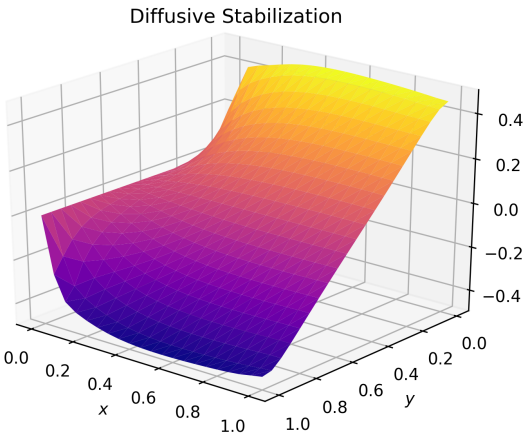


Figure 3.13: Finite element solution for p after one timestep of size $\Delta t = 10^{-2}$, using linear-linear elements and diffusive stabilization.

It is clear that diffusive stabilization does its job well; the solution is completely smooth. More interesting is the behaviour of the bubble stabilization. In the interior of the domain the solution is smooth, but some oscillations occur near the boundary.

3.4 Visco-Poroelasticity on a Deforming Domain

The visco-poroelastic system is given by the following two equations:

$$\rho\left(\frac{D\boldsymbol{v}}{Dt} + (\nabla\cdot\boldsymbol{v})\boldsymbol{v}\right) - \nabla\cdot\boldsymbol{\sigma}(\boldsymbol{u},\boldsymbol{v},p) = \boldsymbol{g} \qquad \text{on } \Omega(t), \tag{3.86}$$

$$\nabla\cdot\boldsymbol{v} - \nabla\cdot(k\nabla p) = f \qquad \text{on } \Omega(t). \tag{3.87}$$

The domain $\Omega(t)$ again depends on the displacement vector $\boldsymbol{u}$ at time t. Furthermore, $\boldsymbol{v}$ is the material derivative of $\boldsymbol{u}$. We impose the following boundary conditions:

$$\boldsymbol{u} = \boldsymbol{0}, \quad k\nabla p\cdot\boldsymbol{n} = 0, \qquad \text{on } \Gamma_1(t), \tag{3.88}$$

$$\boldsymbol{\sigma}\cdot\boldsymbol{n} = \boldsymbol{\tau}, \quad p = 0, \qquad \text{on } \Gamma_2(t), \tag{3.89}$$

The stress tensor is given by

$$\boldsymbol{\sigma}(\boldsymbol{u},\boldsymbol{v},p) = \boldsymbol{\sigma}_{\text{el}}(\boldsymbol{u}) + \boldsymbol{\sigma}_{\text{vis}}(\boldsymbol{v}) - p\boldsymbol{I}, \tag{3.90}$$

where the elastic and viscous parts are given respectively by (2.38) and (2.39).

3.4.1 Derivation of System of Equations

We derive the weak formulations of (3.86) and (3.87). Let $\boldsymbol{\varphi}(\cdot,t)$ be a Lagrangian test field on $\Omega(t)$ that vanishes on $\Gamma_1(t)$. The weak form of equation (3.86) is given by:

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)}\boldsymbol{v}\cdot\boldsymbol{\varphi}\,\mathrm{d}\Omega + \int_{\Omega(t)}\boldsymbol{\sigma}(\boldsymbol{u},\boldsymbol{v}):\boldsymbol{\varepsilon}(\boldsymbol{\varphi})\,\mathrm{d}\Omega - \int_{\Omega(t)}p(\nabla\cdot\boldsymbol{\varphi})\,\mathrm{d}\Omega$$
$$= \int_{\Omega(t)}\boldsymbol{g}\cdot\boldsymbol{\varphi}\,\mathrm{d}\Omega + \int_{\Gamma_2(t)}\boldsymbol{\tau}\cdot\boldsymbol{\varphi}\,\mathrm{d}\Gamma. \tag{3.91}$$

We have used Theorem 3 to simplify the first term of (3.86). Let $\psi(\cdot,t)$ be a scalar-valued test field on $\Omega(t)$ that vanished on $\Gamma_2(t)$. Then the weak form of equation (3.87) is given by:

$$\int_{\Omega(t)}(\nabla\cdot\boldsymbol{v})\psi\,\mathrm{d}\Omega + \int_{\Omega(t)}k\nabla p\cdot\nabla\psi\,\mathrm{d}\Omega = \int_{\Omega(t)}f\psi\,\mathrm{d}\Omega. \tag{3.92}$$

Let $\mathcal{T}_h(t)$ be a triangulation of the approximate domain $\Omega_h(t)$. We use linear basis functions with respect to the triangles in $\mathcal{T}_h(t)$ for $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{p}$. We then obtain the following Galerkin equations corresponding to the weak forms (3.91) and (3.92):

$$\rho\frac{\mathrm{d}}{\mathrm{d}t}\left(M_v\boldsymbol{v}\right) + S_{\text{el}}\boldsymbol{u} + S_{\text{vis}}\boldsymbol{v} - D^\top\boldsymbol{p} = \boldsymbol{g}, \tag{3.93}$$

$$D\boldsymbol{v} + kL\boldsymbol{p} = \boldsymbol{f}. \tag{3.94}$$

Note that $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{p}$ now denote numerical vectors containing the values of their corresponding unknowns in the grid points of $\mathcal{T}_h(t)$. Moreover, the vector $\boldsymbol{g}$ contains both the domain and boundary integral in the right-hand side of (3.91). We write $\boldsymbol{u} = (\boldsymbol{u}^1,\boldsymbol{u}^2)^\top$ and $\boldsymbol{v} = (\boldsymbol{v}^1,\boldsymbol{v}^2)^\top$, then the matrices M_v, S_{el}, S_{vis} and D can be subdivided into blocks:

$$M_v = \begin{pmatrix} M & 0 \\ 0 & M \end{pmatrix}, \quad S_{\text{el}} = \begin{pmatrix} S_{\text{el}}^{11} & S_{\text{el}}^{12} \\ S_{\text{el}}^{21} & S_{\text{el}}^{22} \end{pmatrix}, \quad S_{\text{vis}} = \begin{pmatrix} S_{\text{vis}}^{11} & S_{\text{vis}}^{12} \\ S_{\text{vis}}^{21} & S_{\text{vis}}^{22} \end{pmatrix}, \quad D = \begin{pmatrix} D^1 & D^2 \end{pmatrix}. \tag{3.95}$$

For component-wise definitions of the S_{el} and S_{vis} blocks, we refer to 2.3.4. The divergence blocks are given by (3.67). The matrix L is a Laplacian matrix, it is given by (3.71). Finally the mass matrix M is given element-wise by:

$$M_{ij} = \int_{\Omega(t)}\varphi_i\varphi_j\,\mathrm{d}\Omega, \tag{3.96}$$

where φ_i is a basis function corresponding to the i'th grid point in $\mathcal{T}_h(t)$. We discretize equations (3.93) and (3.94) in time using implicit Euler:

$$\rho M_v^m\boldsymbol{v}^m + \Delta t S_{\text{el}}^m\boldsymbol{u}^m + \Delta t S_{\text{vis}}^m\boldsymbol{v}^m - \Delta t(D^m)^\top\boldsymbol{p}^m = \rho M_v^{m-1}\boldsymbol{v}^{m-1} + \Delta t\boldsymbol{g}^m \tag{3.97}$$

$$D^m\boldsymbol{v}^m + kL^m\boldsymbol{p}^m = \boldsymbol{f}^m. \tag{3.98}$$

The superscript $(\cdot)^m$ denotes a time-level. Note that the coefficient matrices also depend on time because their entries are integrations over the current domain. To eliminate $\boldsymbol{u}^m$ from the system we substitute

$$\boldsymbol{u}^m = \boldsymbol{u}^{m-1} + \Delta t \boldsymbol{v}^m, \tag{3.99}$$

into (3.97) to obtain:

$$\left(\rho M_v^m + \Delta t S_{\text{vis}}^m + \Delta t^2 S_{\text{el}}^m\right) \boldsymbol{v}^m - \Delta t (D^m)^\top \boldsymbol{p}^m = \rho M_v^{m-1} \boldsymbol{v}^{m-1} - \Delta t S_{\text{el}}^m \boldsymbol{u}^{m-1} + \Delta t \boldsymbol{g}^m \tag{3.100}$$

$$D^m \boldsymbol{v}^m + k L^m \boldsymbol{p}^m = \boldsymbol{f}^m. \tag{3.101}$$

In block matrix form we get:

$$\begin{pmatrix} \rho M_v^m + \Delta t S_{\text{vis}}^m + \Delta t^2 S_{\text{el}}^m & -\Delta t (D^m)^\top \\ D^m & k L^m \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} \rho M_v^{m-1} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \Delta t \left(\boldsymbol{g}^m - S_{\text{el}}^m \boldsymbol{u}^{m-1}\right) \\ \boldsymbol{f}^m \end{pmatrix}. \tag{3.102}$$

The above system must be solved using a Picard-type iterative scheme. We use the following convergence criterion:

$$\frac{\|\boldsymbol{v}^{\ell+1} - \boldsymbol{v}^\ell\|}{\|\boldsymbol{v}^\ell\|} + \frac{\|\boldsymbol{p}^{\ell+1} - \boldsymbol{p}^\ell\|}{\|\boldsymbol{p}^\ell\|} \leqslant \delta. \tag{3.103}$$

Here ℓ does *not* denote a time-level, but a sub-iteration within one timestep. We find that choosing δ between between 10^{-4} and 10^{-8} produces good results, depending on the meshwidth.

3.4.2 Stabilization

We observe oscillations in the pressure solution for small values of k. We investigate what is causing this behaviour. Write system (3.102) in the following abbreviated form:

$$\begin{pmatrix} Q & -\Delta t D^\top \\ D & kL \end{pmatrix} \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{p} \end{pmatrix} = \begin{pmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{pmatrix} \tag{3.104}$$

Here we have dropped the time-level superscripts and called the right-hand side vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ respectively. Moreover, $Q = \rho M_v + \Delta t S_{\text{vis}} + \Delta t^2 S_{\text{el}}$. Rearrange the first equation to

$$\boldsymbol{v} = \Delta t Q^{-1} D^\top \boldsymbol{p} + Q^{-1} \boldsymbol{a}. \tag{3.105}$$

Substitute this expression into the second equation to find

$$(kL + \Delta t D Q^{-1} D^\top) \boldsymbol{p} = \boldsymbol{b} - D Q^{-1} \boldsymbol{a}. \tag{3.106}$$

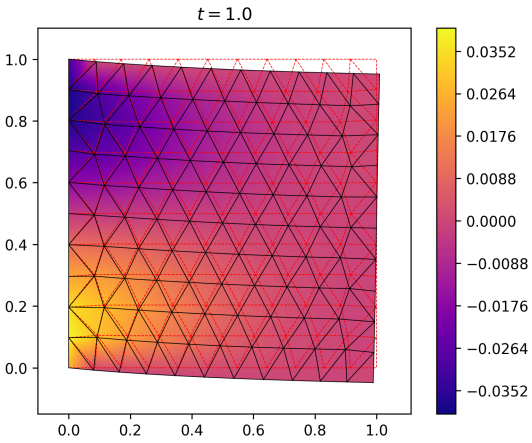
As the oscillations occur for small values of k, it appears that the matrix $\Delta t D Q^{-1} D^\top$ has unfavorable properties. Note that the oscillations can also be fixed by reducing Δt. This solution quickly becomes computationally unfeasible due the high number of time steps necessary. In Chapter 4 we construct a diffusive stabilization scheme that is optimal in a certain sense.

3.4.3 Numerical Experiments

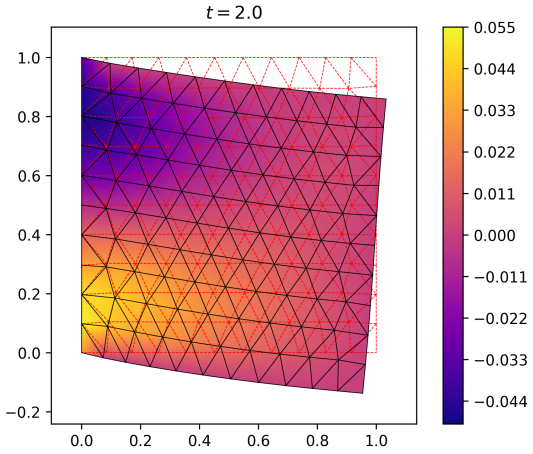
We test the model in a similar situation to the one in section 2.3.6. Let $\Omega(0) = (0,1)^2$ and set $\Gamma_1(0) = \{(x,y) : x = 0\}$. Then from boundary condition (3.89) it is clear that $\Gamma_1(t) = \Gamma_1(0)$ for all $t > 0$. Furtermore, we let $\Gamma_2(t) = \partial\Omega(t) \setminus \Gamma_1(t)$. The body force is set to $\boldsymbol{g} = (0, -0.1)^\top$, the pressure source is set to zero: $f = 0$, and the force on $\Gamma_2(t)$ is also set to zero: $\boldsymbol{\tau} = \boldsymbol{0}$. The following parameter values are used:

$$\rho = 1, \quad \lambda_\varepsilon = \mu_\varepsilon = \lambda_v = \mu_v = 1, \quad k = 10^{-2}. \tag{3.107}$$

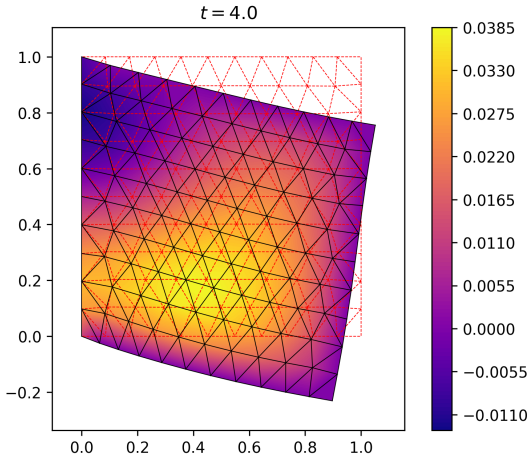
We use a meshwidth of $h = 0.1$ and a timestep of $\Delta t = 0.05$. Figures 3.14a - 3.14d show the displacement and pressure solution at various timesteps. Note that the pressure is shown as a contour plot superimposed on the displaced mesh. The initial mesh is shown for reference in red.



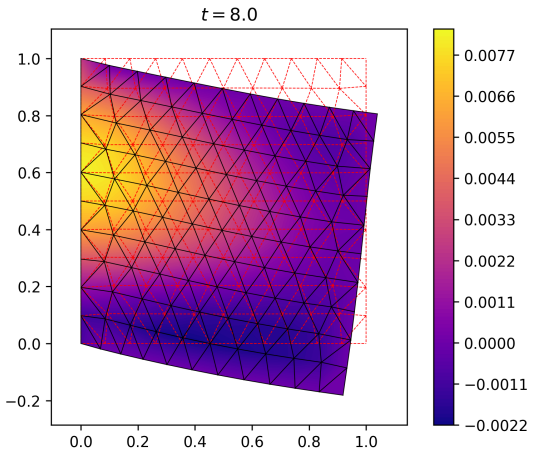
(a) Finite element solutions at $t = 1.0$.



(b) Finite element solutions at $t = 2.0$.



(c) Finite element solutions at $t = 4.0$.



(d) Finite element solutions at $t = 8.0$.

Figure 3.14: Plots showing the finite element solution at various points in time.

Note that the mesh displacement looks very similar to those in Figures 2.6a - 2.6d. This is due to a relatively high values of k: the pressure does not have a big effect on the displacement (velocity). Figure 3.15 shows the finite element solutions at $t = 8$ for a lower value $k = 10^{-5}$, whereas the other parameters are kept constant. Compared to Figures 3.14d and 2.6d, we can see a small difference in displacement. Note that lowering the value of k even further will result in more severe oscillations. They can already be seen forming near the top and bottom left corners.

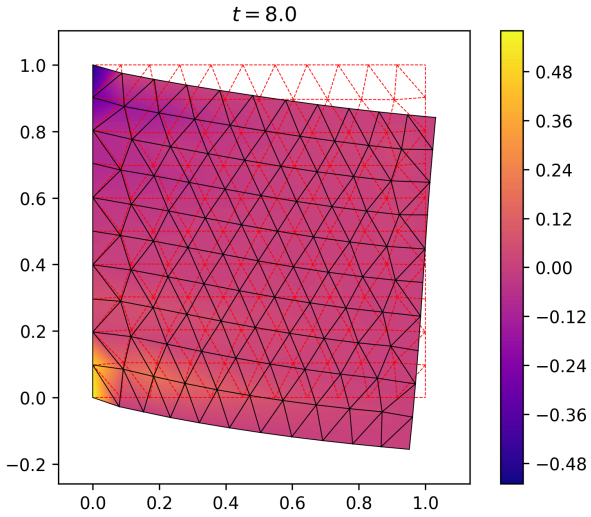


Figure 3.15: Finite element solutions at $t = 8$ for $k = 10^{-5}$

3.5 Comparison Between Viscoelastic and Visco-Poroelastic Models

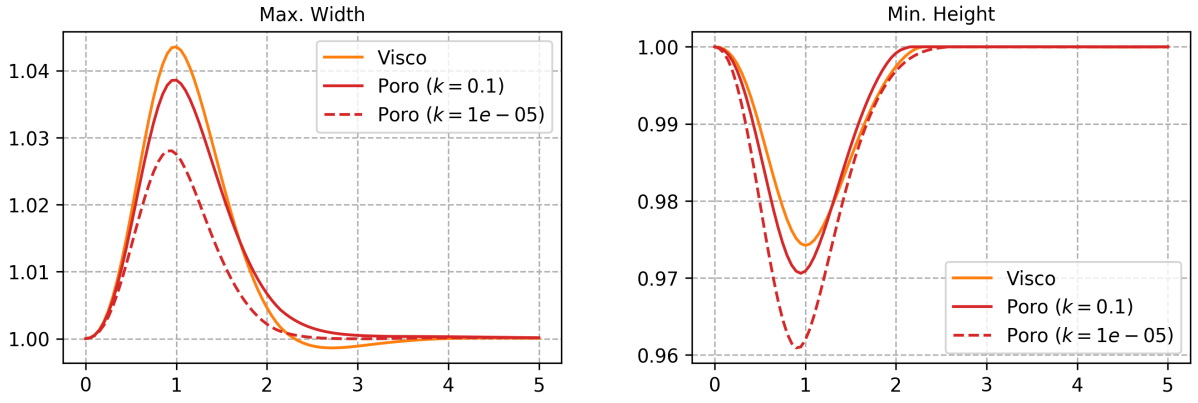
We compare the behaviour of the visco-poroelastic model from Section 3.4 to the viscoelastic model from Section 2.3. We again apply the body force

$$\boldsymbol{g}(\boldsymbol{x}, t) = \begin{pmatrix} 0.2\sin(\pi t)\mathbb{1}_{[0,1]}(t) \\ 0 \end{pmatrix} \tag{3.108}$$

to the unite square like in Section 2.5. The mass source function f is set to zero, and the following mechanical parameters are used:

$$\mu_\varepsilon = \lambda_\varepsilon = 1, \quad \mu_v = \lambda_v = 0.5, \quad \rho = 0.5. \tag{3.109}$$

We use the (initially) uniform finite element mesh from Figure 2.10 and a timestep of size $\Delta t = 0.05$. Figures 3.16a and 3.16b show the maximum width and minimum height of the domain respectively as functions of time. Two different values of k are used. For $k = 10^{-5}$, the stabilized model is used. One can clearly see that the inclusion of poroelasticity has an effect on the evolution of the domain. This effect is more pronounced for smaller values of k.



(a) Maximum width of domain.

(b) Minimum height of domain.

Figure 3.16: Maximum domain width and mimimum domain height as a function of time for two visco-poroelastic models compared to the viscoelastic model.

We give an intuitive explanation from a physical point of view for the effect of k on the solution. Recall that k is the quotient of the permeability of the medium and the dynamic viscosity of the poro-fluid. If

k is large, then either the medium is very permeable of the fluid has a low viscosity. In both cases, the fluid is able to move relatively freely. Thus, whenever the medium is deformed, the fluid will quickly adapt to the new situation. On the other hand, if k is small, either the permeability is low or the fluid has a high viscosity. In this case the fluid has more trouble to move through the medium, hence it will react slower to deformations. This inability of the poro-fluid to adapt to the new situation causes it have a more pronounced effect on the displacement.

Part II

4

Morpho-Poroelasticity and Analysis of Stabilized Finite Element Methods

In this chapter we present new work. We first analyze the properties of diffusive stabilization applied to poroelastic systems. The aim is to give a rigorous derivation of the optimal tuning parameter β. This derivation is performed in Sections 4.2, 4.3 and 4.4 for a one-dimensional visco-poroelastic system. For a two- or higher dimensional systems, the computations become too hard to do exactly. For this reason, the result obtained in the one-dimensional case is 'extrapolated' to higher dimensional poroelastic systems. The second part of this chapter contains the construction of a finite element model for a morpho-poroelastic system. Diffusive stabilization from earlier sections is also applied to this model, since the problematic equation has a very similar structure.

4.1 Stabilizing Poroelastic Systems

For small k, the visco- poroelastic system as presented in (3.102), as well as the morpho-poroelastic system (4.76) will start to show non-physical oscillations in the pressure solution. To eliminate them a diffusive stabilization term can be added to the pressure equation, such as we have done for the poroelastic system in Section 3.3.3. Note that in both the visco-poroelastic and morpho-poroelastic systems the last equation in the fully discretized system is given by

$$D^m \boldsymbol{v}^m + kL^m \boldsymbol{p}^m = \boldsymbol{f}^m. \tag{4.1}$$

As we have seen in Section 3.3.3, the idea is to introduce a small diffusive pertubation. This causes the solution to remain smooth. It is possible to only add this pertubation to the left-hand side of (4.1), effectively increasing k. Equation (4.1) is then replaced by

$$D^m \boldsymbol{v}^m + (k+\beta)L^m \boldsymbol{p}^m = \boldsymbol{f}^m. \tag{4.2}$$

Another option would be to add a pertubation to both sides, thereby approximating some kind of time derivative of the Laplacian of p. In this case, equation (4.1) becomes

$$D^m \boldsymbol{v}^m + (k+\beta)L^m \boldsymbol{p}^m = \boldsymbol{f}^m + \beta L^{m-1} \boldsymbol{p}^{m-1}. \tag{4.3}$$

Both (4.2) and (4.3) will have a similar effect on the stability of the solution, since that is determined by the matrix $(k+\beta)L$. The two approaches differ from each other in the right-hand side, and therefore will lead to different errors compared to (4.1). To gain insight in the error, we determine the differential equations corresponding to (4.2) and (4.3). Recall that (4.1) is the discretized version of the differential equation

$$\nabla \cdot \boldsymbol{v} - k\nabla^2 p = f. \tag{4.4}$$

Thus, we can immediately tell that (4.2) discretizes the following perturbed differential equation:

$$\nabla \cdot \boldsymbol{v} - k\nabla^2 p = f + \beta\nabla^2 p. \tag{4.5}$$

The corresponding differential equation of (4.3) is less obvious. See the following theorem:

Theorem 6. *Assume that $\boldsymbol{v}$ and $\boldsymbol{p}$ satisfy*

$$\nabla \cdot \boldsymbol{v} - k\nabla^2 p = f + \Delta t\beta \left(\frac{D(\nabla^2 p)}{Dt} + (\nabla \cdot \boldsymbol{v})(\nabla^2 p) \right). \tag{4.6}$$

Then the corresponding finite element solution vectors $\boldsymbol{v}^m$ and $\boldsymbol{p}^m$ satisfy (4.3)*, provided Lagrangian basis functions are used and the time discretization is done using implicit Euler.*

Proof. Assume we have homogeneous boundary conditions on p, such as in (3.88)-(3.89). Let q be a Lagrangian basis function. Then the weak form of (4.6) is given by

$$\int_{\Omega(t)} (\nabla \cdot \boldsymbol{v}) q \, \mathrm{d}\Omega + \int_{\Omega(t)} k\nabla p \cdot \nabla q \, \mathrm{d}\Omega = \int_{\Omega(t)} fq \, \mathrm{d}\Omega + \Delta t\beta \int_{\Omega(t)} \left(\frac{D(\nabla^2 p)}{Dt} + (\nabla \cdot \boldsymbol{v})(\nabla^2 p) \right) q \, \mathrm{d}\Omega.$$

The last term can be simplified using Theorem 3:

$$\int_{\Omega(t)} (\nabla \cdot \boldsymbol{v}) q \, \mathrm{d}\Omega + \int_{\Omega(t)} k\nabla p \cdot \nabla q \, \mathrm{d}\Omega = \int_{\Omega(t)} fq \, \mathrm{d}\Omega + \Delta t\beta \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} (\nabla^2 p) q \, \mathrm{d}\Omega.$$

By applied partial integration once more, we obtain

$$\int_{\Omega(t)} (\nabla \cdot \boldsymbol{v}) q \, \mathrm{d}\Omega + \int_{\Omega(t)} k\nabla p \cdot \nabla q \, \mathrm{d}\Omega = \int_{\Omega(t)} fq \, \mathrm{d}\Omega - \Delta t\beta \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \nabla p \cdot \nabla q \, \mathrm{d}\Omega.$$

By computing the Galerkin equations, the semi-discretized system is obtained:

$$D\boldsymbol{v} + kL\boldsymbol{p} = \boldsymbol{f} - \Delta t\beta \frac{\mathrm{d}}{\mathrm{d}t}(L\boldsymbol{p}),$$

where D is the divergence matrix and L is the Laplacian matrix. Applying implicit Euler yields

$$D^m \boldsymbol{v}^m + kL^m \boldsymbol{p}^m = \boldsymbol{f}^m - \beta \left(L^m \boldsymbol{p}^m - L^{m-1} \boldsymbol{p}^{m-1} \right).$$

Equation (4.3) is obtained by rearranging terms. □

To summarize, we have found that adding a one-sided stabilization to the fully discretized system such as in (4.2) is equivalent to solving (4.5) using finite elements. Similarly, adding the two-sided stabilization such as in (4.3) amounts to solving (4.6). Assume the system converges to a steady state, thus $\boldsymbol{v} \to \boldsymbol{0}$ as $t \to \infty$. Then the stationary pressure equations corresponding to both (4.4) and (4.6) become

$$-k\nabla^2 p = f. \tag{4.7}$$

Whereas for (4.5) it becomes

$$-(k+\beta)\nabla^2 p = f. \tag{4.8}$$

Based on this, we argue that a two-sided stabilization is the way to go.

4.2 Analysis of One-Dimensional Visco-Poroelastic System

In both (4.2) and (4.3) the stability of the solution is affected by the tuning parameter β. If β is too small the solution will still contain oscillations. On the other hand, if β is too big the solution will be excessively smoothened, leading to a large error. In the following sections we derive an optimal tuning parameter β for a one-dimensional visco-poroelastic system on a fixed, uniform grid. This optimal choice of β is valid in the limit of $h \to 0$, where h is the meshwidth. In this framework system (3.102) becomes

$$\begin{pmatrix} \rho M_v + (\Delta t(\mu_v + \lambda_v) + \Delta t^2(\mu_\varepsilon + \lambda_\varepsilon))L_v & -\Delta t D^\top \\ D & kL_p \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^m \\ \boldsymbol{p}^m \end{pmatrix} = \begin{pmatrix} \rho M_v & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^{m-1} \\ \boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \Delta t(\boldsymbol{g}^m - L_v \boldsymbol{u}^{m-1}) \\ \boldsymbol{f}^m \end{pmatrix}. \tag{4.9}$$

where S_{vis} is replaced with $(\mu_v + \lambda_v)L_v$ and S_{el} with $(\mu_\varepsilon + \lambda_\varepsilon)L_v$, where L_v is a Laplace matrix encorporating the boundary conditions imposed on v. We define

$$\theta = \Delta t(\mu_v + \lambda_v) + \Delta t^2(\mu_\varepsilon + \lambda_\varepsilon), \quad \text{and} \quad A = \rho M_v + \theta L_v. \tag{4.10}$$

By isolating $\boldsymbol{v}^m$ in the first equation of (4.9) and substituting the result into the second, we obtain the following evolution equation for $\boldsymbol{p}^m$:

$$\left(kL_p + \Delta t D A^{-1} D^\top\right) \boldsymbol{p}^m = \boldsymbol{f}^m - \rho D A^{-1} M_v \boldsymbol{v}^{m-1} + \Delta t D A^{-1} \left(L_v \boldsymbol{u}^{m-1} - \boldsymbol{g}^m\right). \tag{4.11}$$

Let $C = DA^{-1}D^\top$, then the coefficient matrix $kL_p + \Delta t C$ is important: it causes the notorious oscillations in p. From numerical tests we know that oscillations only occur for small values of k. Since L_p is an M-matrix, the matrix $\Delta t D A^{-1} D^\top$ must be the cause of the oscillations. Equation (4.11) shows that the effect of the problematic matrix can be reduced by decreasing Δt. Numerical tests show that the system is again stable if we have $\Delta t \leqslant k$ whenever k is small. In this section the matrix C is approximated. With this approximation an approximate optimal tuning parameter β can be determined.

4.3 Approximating the Inverse of the Discrete Reaction-Laplacian Operator

The inverse of A is approximated for a certain set of boundary conditions. To this end, we consider a related boundary value problem with a delta function as source. This problem can be solved exactly but also using finite elements. By using the latter method we encounter simplified versions of the matrices M_v and L_v. Comparing the exact solution to the finite element solution allows us to approximate A^{-1}. Consider the following reaction-Laplacian equation and corresponding boundary conditions:

$$\rho u - \theta u'' = \delta(x - \alpha), \quad x \in [0, 1] \tag{4.12}$$

$$u'(0) = 0, \tag{4.13}$$

$$u(1) = 0, \tag{4.14}$$

where $\delta(\cdot)$ denotes the Dirac delta-function, and $\alpha \in [0, 1)$. Moreover, the parameters ρ and θ are strictly positive. Using the Laplace transform, the following exact (weak) solution is obtained:

$$\begin{aligned} u(x) &= \frac{1}{\sqrt{\rho\theta}} \left[\frac{\sinh\left(\lambda(1-\alpha)\right)}{\cosh\left(\lambda\right)} \cosh\left(\lambda x\right) - H(x-\alpha) \sinh\left(\lambda(x-\alpha)\right) \right] \\ &= \frac{\lambda}{\rho \cosh(\lambda)} \begin{cases} \sinh(\lambda(1-\alpha)) \cosh(\lambda x), & x \leqslant \alpha \\ \sinh(\lambda(1-x)) \cosh(\lambda\alpha), & x > \alpha \end{cases}, \end{aligned} \tag{4.15}$$

where $\lambda = \sqrt{\rho/\theta}$. Note that (4.15) is not a classical solution, since it is not twice continuously differentiable on $(0, 1)$. Figure 4.1 shows a plot of this solution for certain values of α, ρ and θ.

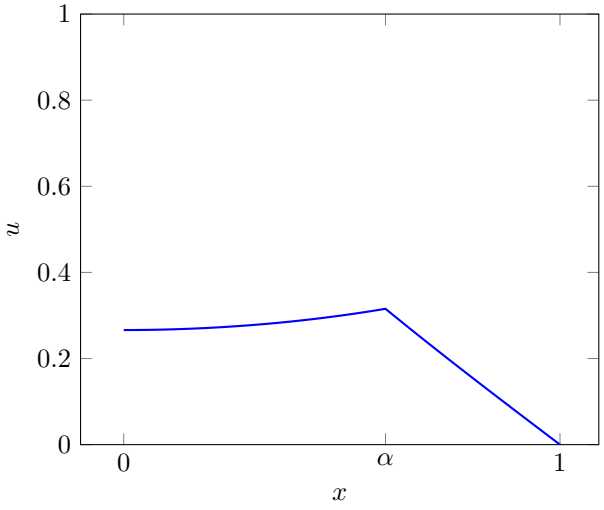


Figure 4.1: Plot of exact solution as given in (4.15) for $\alpha = 0.6$ and $\rho = \theta = 1$.

4.3.1 Existence and Uniqueness of Weak Solution to Reaction-Laplacian Equation

We prove that a solution to the reaction-Laplacian problem (4.12)-(4.14) exists in the weak sense. Define the function space

$$H_0^1(0, 1) = \{f \in H^1(0, 1) : f(1) = 0\}, \tag{4.16}$$

which inherits its norm from $H^1(0,1)$:

$$\|f\|_{H^1}^2 = \|f\|_{L^2}^2 + \|f'\|_{L^2}^2 = \int_0^1 f^2 + (f')^2 \,\mathrm{d}x.$$

The weak form corresponding to problem (4.12)-(4.14) is given by:

Find $u \in H_0^1(0,1)$ such that for all $v \in H_0^1(0,1)$ we have:

$$a(u,v) = \delta_\alpha(v), \tag{4.17}$$

where

$$a(u,v) = \int_0^1 \rho uv + \theta u'v' \,\mathrm{d}x, \quad \text{and} \quad \delta_\alpha(v) = v(\alpha). \tag{4.18}$$

To show that a solution can be obtained using the finite element method, it must first be shown that the weak form (4.17) has a unique solution. This is summarized in the following theorem:

Theorem 7. *There exists a unique $u \in H_0^1(0,1)$ that satisfies*

$$a(u,v) = \delta_\alpha(v)$$

for all $v \in H_0^1(0,1)$, where $a(\cdot,\cdot)$ and $\delta_\alpha(\cdot)$ are given in (4.18)*.*

Proof. The statement can be proved using the Lax-Milgram theorem. First, the necessary conditions on a are proved. Note that we view $a(\cdot,\cdot)$ as the bilinear form

$$a : H_0^1(0,1) \times H_0^1(0,1) \to \mathbb{R}.$$

It is clear from its definition that a is symmetric. Furthermore,

- a is bounded (and hence continuous). Indeed:

$$\begin{aligned}
|a(u,v)| &= \left| \int_0^1 \rho uv + \theta u'v' \,\mathrm{d}x \right| \\
&\leqslant \rho \int_0^1 |uv| \,\mathrm{d}x + \theta \int_0^1 |u'v'| \,\mathrm{d}x \\
&\leqslant \rho \sqrt{\int_0^1 u^2 \,\mathrm{d}x} \sqrt{\int_0^1 v^2 \,\mathrm{d}x} + \theta \sqrt{\int_0^1 (u')^2 \,\mathrm{d}x} \sqrt{\int_0^1 (v')^2 \,\mathrm{d}x} \\
&= \rho \|u\|_{L^2} \|v\|_{L^2} + \theta \|u'\|_{L^2} \|v'\|_{L^2} \\
&\leqslant (\rho + \theta) \|u\|_{H^1} \|v\|_{H^1}.
\end{aligned}$$

  In the third step the Cauchy-Schwarz inequality is applied.

- a is coercive:

$$\begin{aligned}
a(u,u) &= \int_0^1 \rho u^2 + \theta (u')^2 \,\mathrm{d}x \\
&= \rho \|u\|_{L^2}^2 + \theta \|u'\|_{L^2}^2 \\
&\geqslant \min\{\rho, \theta\} \left(\|u\|_{L^2}^2 + \|u'\|_{L^2}^2 \right) \\
&= \min\{\rho, \theta\} \|u\|_{H^1}^2.
\end{aligned}$$

In order for Lax-Milgram to be applicable, the right-hand side of (4.17) must be continuous. Note that we view $\delta_\alpha(\cdot)$ as a functional acting on $H_0^1(0,1)$. Before continuity is proved, the issue of δ_α being well-defined must be addressed. After all, since $v \in H_0^1(0,1) \subset L^2(0,1)$, pointwise evaluation of v is not defined. Let (a,b) be a real interval, then we have the inclusion[1] $H^1(a,b) \subset AC(a,b)$, where $AC(a,b)$ is the set of absolutely continuous functions on (a,b). A proof of this fact is provided in [2]. A well known

[1]The inclusion holds almost everywhere: for each $f \in H^1(a,b)$ there is an $\widetilde{f} \in AC(a,b)$ such that $f = \widetilde{f}$ a.e.

property of absolutely continuous functions is that the fundamental theorem of integration holds [9]. Let $f \in AC(a,b)$, then for $x \in (a,b)$:

$$f(x) = f(a) + \int_a^x f'(t)\,\mathrm{d}t,$$

where f' is a weak derivative of f. We can now prove the continuity of δ_α. It suffices to show δ_α is bounded due to its linearity. Let $u \in H_0^1(0,1)$, then we have

$$|\delta_\alpha(u)| = |u(\alpha)| = \left| u(0) + \int_0^\alpha u'\,\mathrm{d}x \right|$$

To eliminate $u(0)$, note that $u(1) = 0$, and therefore:

$$u(0) + \int_0^1 u'\,\mathrm{d}x = 0.$$

It follows that:

$$|\delta_\alpha(u)| = \left| \int_0^\alpha u'\,\mathrm{d}x - \int_0^1 u'\,\mathrm{d}x \right| = \left| \int_\alpha^1 u'\,\mathrm{d}x \right| \leqslant \int_\alpha^1 |u'|\,\mathrm{d}x$$

Apply the Cauchy-Schwarz inequality:

$$\begin{aligned}
\int_\alpha^1 |u'|\,\mathrm{d}x &\leqslant \sqrt{1-\alpha}\,\sqrt{\int_\alpha^1 (u')^2\,\mathrm{d}x} \\
&\leqslant \sqrt{1-\alpha}\,\sqrt{\int_0^1 (u')^2\,\mathrm{d}x} \\
&= \sqrt{1-\alpha}\,\|u'\|_{L^2} \\
&\leqslant \sqrt{1-\alpha}\,\|u\|_{H^1}
\end{aligned}$$

Which completes the proof of the boundedness of δ_α. All conditions of the Lax-Milgram theorem are met, thus problem (4.17) has a unique solution. □

4.3.2 Error Analysis

In finite elements, the solution- and test space (which are equal in our case) are replaced by finite dimensional spaces. Let $V = H_0^1(0,1)$ be the original test space, and let V_h be a finite dimensional subspace of V. Note that V_h will be specified later on. In the remainder of this section, we derive the L^2-error of the finite element solution. Assume that $u \in V$ satisfies

$$a(u,v) = \delta_\alpha(v), \qquad \forall v \in V, \tag{4.19}$$

and $u_h \in V_h$ satisfies

$$a(u_h, v_h) = \delta_\alpha(v_h), \qquad \forall v_h \in V_h. \tag{4.20}$$

It follows from the proof of Theorem 7 that $a(\cdot,\cdot)$ defines an inner product on V. We say that $v, w \in V$ are a-orthogonal if $a(v,w) = 0$. Note that the finite element error $u - u_h$ is a-orthogonal to the test space V_h:

$$a(u - u_h, v_h) = 0, \qquad \forall v_h \in V_h. \tag{4.21}$$

Indeed:

$$\begin{aligned}
a(u - u_h, v_h) &= a(u, v_h) - a(u_h, v_h) \\
&= \delta_\alpha(v_h) - \delta_\alpha(v_h) = 0
\end{aligned}$$

Before the L^2 norm of the error can be computed, its energy norm is analyzed.

Definition 4. The a-induced *energy norm* is defined by:

$$\|u\|_a^2 = a(u,u) \tag{4.22}$$

We have the following well-known result:

Lemma 5 (Céa)**.** *Let $u \in V$ and $u_h \in V_h$ satisfy* (4.19) *and* (4.20) *respectively. Then:*

$$\|u - u_h\|_a \leqslant \|u - v_h\|_a, \qquad \forall v_h \in V_h. \tag{4.23}$$

Proof. Let $v_h \in V_h$ be arbitrary. The proof follows from the a-orthogonality property of $u - u_h$:

$$\begin{aligned}
\|u - u_h\|_a^2 &= a(u - u_h, u - u_h) \\
&= a(u - u_h, u - v_h + v_h - u_h) \\
&= a(u - u_h, u - v_h) + a(u - u_h, u_h - v_h) \\
&= a(u - u_h, u - v_h) \\
&\leqslant \|u - u_h\|_a \|u - v_h\|_a
\end{aligned}$$

We have used the fact that $v_h - u_h \in V_h$. In the last step, the Cauchy-Schwarz inequality is applied. Dividing both sides by $\|u - u_h\|_a$ yields the desired inequality. □

Céa's Lemma tells us that u_h is the best approximation to u in the energy norm. It also allows for an error bound in the H^1-norm:

Corollary 1. *Let $u \in V$ and $v_h \in V_h$ satisfy* (4.19) *and* (4.20) *respectively. Then:*

$$\|u - u_h\|_{H^1} \leqslant \sqrt{\frac{\rho + \theta}{\min\{\rho, \theta\}}}\, \|u - v_h\|_{H^1}, \qquad \forall v_h \in V_h. \tag{4.24}$$

Proof. Let $v_h \in V_h$ be arbitrary. Using the coercivity and boundedness property of a (see the proof of Theorem 7), and Céa's Lemma, we have:

$$\begin{aligned}
\min\{\rho, \theta\}\|u - u_h\|_{H^1}^2 &\leqslant a(u - u_h, u - u_h) \\
&= \|u - u_h\|_a^2 \\
&\leqslant \|u - v_h\|_a^2 \\
&= a(u - v_h, u - v_h) \\
&\leqslant (\rho + \theta)\|u - v_h\|_{H^1}^2.
\end{aligned}$$

Rearrange terms and take the square root on both sides to obtain the desired inequality. □

Céa's Lemma and its corollary tell us that that the finite element solution u_h is the 'best' in some sense. We can exploit this property by constructing a function in V_h that approximates u really well. By Céa's Lemma, u_h will be an even better approximation (in the energy norm). We will use Lagrangian linear interpolation polynomials.

Lemma 6 (Linear Interpolation)**.** *Let $\mathcal{T}_h$ be a mesh on $(0, 1)$, that is:*

$$\mathcal{T}_h = \{T_i = (x_i, x_{i+1}) \mid i \in \{1, \ldots, n\}\}, \qquad \text{such that} \quad \bigcup_{i=1}^{n} \overline{T_i} = (0, 1).$$

Let $w \in H^1(0, 1)$ such that for each $T \in \mathcal{T}_h$ we have $w \in C^2(T)$ and w_i'' is bounded on T. Here we use the following shorthand: $w_i := w|_{T_i}$. Let $\pi_h w$ be the piecewise linear interpolant of w such that $\pi_h w(x_i) = w(x_i)$ for each grid point x_i. Then there exists some $c > 0$ such that:

$$\|w - \pi_h w\|_{H^1} \leqslant cKh,$$

where

$$h = \max_i |T_i|,$$

and

$$K = \max_i\{K_i\}, \qquad \text{for} \qquad K_i = \sup_{x \in T_i} \left\{w_i''(x)^2\right\}. \tag{4.25}$$

Proof. Define the interpolation error $e(x) = w(x) - \pi_h w(x)$. Consider the element $T_i = (x_i, x_{i+1})$. Since $e(x_i) = e(x_{i+1}) = 0$, by Rolle's theorem there is a $\xi \in T_i$ such that $e'(\xi) = 0$. Taking into account that $(\pi_h w)''(x) = 0$ on T_i, we get:

$$e'(x) = e'(x) - e'(\xi) = \int_\xi^x e''(t)\,\mathrm{d}t = \int_\xi^x w_i''(t)\,\mathrm{d}t.$$

By the Cauchy-Schwarz inequality we find for $x \in T_i$:

$$|e'(x)| \leqslant \int_\xi^x |w_i''(t)|\,\mathrm{d}t \leqslant \int_{T_i} |w_i''(t)|\,\mathrm{d}t \leqslant \sqrt{|T_i|}\sqrt{\int_{T_i} w_i''(t)^2\,\mathrm{d}t}.$$

It follows that

$$\begin{aligned}
\|e'\|_{L^2}^2 &= \int_0^1 e'(x)^2\,\mathrm{d}x \\
&= \sum_i \int_{T_i} e'(x)^2\,\mathrm{d}x \\
&\leqslant \sum_i \int_{T_i} |T_i| \int_{T_i} w_i''(t)^2\,\mathrm{d}t\,\mathrm{d}x \\
&= \sum_i |T_i|^2 \int_{T_i} (w_i''(t))^2\,\mathrm{d}t \\
&\leqslant \max_i |T_i|^2 \sum_i \int_{T_i} w_i''(t)^2\,\mathrm{d}t
\end{aligned}$$

To eliminate the sum in the right-hand side, note that

$$\sum_i \int_{T_i} w_i''(t)^2\,\mathrm{d}t \leqslant \sum_i \int_{T_i} K_i\,\mathrm{d}t \leqslant \sum_i \int_{T_i} K\,\mathrm{d}t = K.$$

And thus we get:

$$\|e'\|_{L^2}^2 \leqslant Kh^2.$$

The L^2-norm of e can be computed in a similar fashion. Note that for $x \in T_i$:

$$e(x) = e(x) - e(x_i) = \int_{x_i}^x e'(t)\,\mathrm{d}t.$$

And thus by Cauchy-Schwarz:

$$|e(x)| \leqslant \int_{x_i}^x |e'(t)|\,\mathrm{d}t \leqslant \int_{T_i} |e'(t)|\,\mathrm{d}t \leqslant \sqrt{|T_i|}\sqrt{\int_{T_i} e'(t)^2\,\mathrm{d}t}.$$

It follows that:

$$\begin{aligned}
\|e\|_{L^2}^2 &= \int_0^1 e(x)^2\,\mathrm{d}x \\
&= \sum_i \int_{T_i} e(x)^2\,\mathrm{d}x \\
&\leqslant \sum_i \int_{T_i} |T_i| \int_{T_i} e'(t)^2\,\mathrm{d}t\,\mathrm{d}x \\
&\leqslant h^2 \int_0^1 e'(t)^2\,\mathrm{d}t \\
&= h^2 \|e'\|_{L^2}^2 \leqslant Kh^4.
\end{aligned}$$

Having computed both the L^2-norm of e and e', the H^1 can now be determined:

$$\|e\|_{H^1}^2 = \|e\|_{L^2}^2 + \|e'\|_{L^2}^2 \leqslant (h^4 + h^2)K.$$

Note that in our case $h < 1$, and thus $h^4 + h^2 \leqslant 2h^2$. Let $c = \sqrt{2}$, then we conclude that

$$\|e\|_{H^1} \leqslant cKh.$$

□

Note that our bilinear form $a(\cdot, \cdot)$ is uniformly elliptic. Assume w satisfies

$$a(w, v) = (f, v), \qquad \forall v \in V, \tag{4.26}$$

for some $f \in L^2(0, 1)$. Then by the uniform ellipticity of a, we have $w \in H^2(0, 1)$ and

$$\|w\|_{H^2} \leqslant c\|f\|_{L^2}. \tag{4.27}$$

Since $w \in H^2(0, 1)$ the proof of Lemma 6 can be slightly adapted to obtain

$$\|w - \pi_h w\|_{H^1} \leqslant c_1 h \|w''\|_{L^2}.$$

From which it follows that

$$\|w - \pi_h w\|_{H^1} \leqslant c_1 h \|w''\|_{L^2} \leqslant c_2 h \|w\|_{H^2} \leqslant c_3 h \|f\|_{L^2}. \tag{4.28}$$

Note that the constants c_1, c_2 and c_3 do not depend on h. This leads us to the following theorem.

Theorem 8 (L^2-error of FE-solution)**.** *Let $\mathcal{T}_h$ be a mesh on $(0, 1)$ such that α coincides with a grid node. Let $V_h \subset V = H_0^1(0, 1)$ be the space of piecewise linear functions w.r.t. $\mathcal{T}_h$. Assume $u \in V$ is the solution to (4.19), then u is also given by (4.15). Let $u_h \in V_h$ satisfy the finite element problem (4.20), then:*

$$\|u - u_h\|_{L^2} \leqslant cKh^2 \tag{4.29}$$

for some $c > 0$, where $h = \max_{T \in T_h} |T|$ and K is given by (4.25) for $w = u$.

Proof. We use Nitsche's trick to prove the error bound. To this end, the dual problem is introduced. Assume $\varphi \in V$ satisfies

$$a(\varphi, v) = (u - u_h, v), \qquad \forall v \in V.$$

Where a is the same bilinear form as in (4.19) and $(\cdot, \cdot)$ denotes the L^2 inner product. Let $\pi_h \varphi \in V_h$ be the linear interpolant of φ such that $\pi_h \varphi(x_i) = \varphi(x_i)$ for each grid node x_i. Then, by the definition of the dual problem and the a-orthogonality of $u - u_h$ to V_h, we have

$$\begin{aligned} \|u - u_h\|_{L^2}^2 &= (u - u_h, u - u_h) \\ &= a(\varphi, u - u_h) \\ &= a(\varphi - \pi_h \varphi, u - u_h) \\ &\leqslant \|\varphi - \pi_h \varphi\|_a \|u - u_h\|_a. \end{aligned}$$

In the last step, we have applied Cauchy-Schwarz. In the proof of Corollary 1 it is shown that the energy norm is bounded by a constant times the H^1-norm. Then by (4.28) we get:

$$\|\varphi - \pi_h \varphi\|_a \leqslant c_1 \|\varphi - \pi_h \varphi\|_{H^1} \leqslant c_2 h \|\varphi''\|_{L^2} \leqslant c_3 h \|u - u_h\|_{L^2}.$$

The reason that $\|u - u_h\|_{L^2}$ appears is because it is the right-hand side of the dual problem solved by φ. By Céa's Lemma 5 we know that $\|u - u_h\|_a \leqslant \|u - \pi_h u\|_a$, where $\pi_h u$ is the linear interpolant of u w.r.t. $\mathcal{T}_h$. Because we let α coincide with a grid node of $\mathcal{T}_h$, u meets the conditions of Lemma 6. Thus we get the following inequality:

$$\|u - u_h\|_a \leqslant \|u - \pi_h u\|_a \leqslant c_4 \|u - \pi_h u\|_{H^1} \leqslant c_5 Kh.$$

To summarize, we have found

$$\|u - u_h\|_{L^2}^2 \leqslant cKh^2 \|u - u_h\|_{L^2}.$$

Where $c = c_3 c_5$. Divide both sides by $\|u - u_h\|_{L^2}$ to obtain the desired error bound. □

4.3.3 Computing the Pseudo-Inverse

The inverse of the matrix A is approximated. We first build the Galerkin equations corresponding to the finite element formulation of the reaction-Laplacian problem. We will see that the resulting coefficient matrix is very similar to the one in (4.10). Using the error analysis of the previous section, we can construct an approximate inverse to A that is order $h^{3/2}$ accurate per entry. Let $\mathcal{T}_h$ be a uniform mesh on $(0,1)$:

$$\mathcal{T}_h = \{(x_i, x_{i+1}) \mid i \in \{0, \dots, n-1\}\}, \tag{4.30}$$

where $x_i = ih$ and $h = 1/n$. We write $x_i = ih$. Note that the unknowns are located at $x_0, x_1, \dots, x_{n-1}$, but not at $x_n = 1$ as a Dirichlet boundary condition is imposed here. The unknowns are denoted by $u_0, u_1, \dots, u_{n-1}$ respectively. In order for the error analysis of the previous section to be valid, we assume α coincides with one of the grid nodes. Say $\alpha = x_{i_\alpha}$ for some $i_\alpha \in \{0, \dots, n-1\}$. The finite element scheme amounts to solving the linear system

$$A\boldsymbol{u} = \boldsymbol{e}_{i_\alpha}. \tag{4.31}$$

where $A = \rho M + \theta L$. Moreover, $\boldsymbol{u} = (u_0, u_1, \dots, u_{n-1})^\top$ is the vector containing the unknowns and $\boldsymbol{e}_{i_\alpha}$ is the standard basis vector[2] containing a 1 in its i_α'th position. The Laplace matrix L and the mass matrix M are given by

$$L = \frac{1}{h}\begin{pmatrix} 1 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix}, \quad M = \frac{h}{6}\begin{pmatrix} 2 & 1 & & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ & & & & 1 & 4 \end{pmatrix}. \tag{4.32}$$

Note that M and L correspond to M_v and L_v in (4.9) respectively. They are equal whenever the boundary conditions imposed on v are the same as in (4.13)-(4.14). We let i_α cycle through every posible index to obtain the n different linear systems

$$A\boldsymbol{u}_i = \boldsymbol{e}_i, \qquad i \in \{0, \dots, n-1\}. \tag{4.33}$$

It follows that

$$\boldsymbol{u}_i = A^{-1}\boldsymbol{e}_i. \tag{4.34}$$

Thus, $\boldsymbol{u}_i$ is the i'th column of A^{-1}. Note that A is symmetric, hence A^{-1} is also symmetric. Let $u_i(x)$ be the exact solution given by (4.15) where we choose $\alpha = x_i$. Since $(\boldsymbol{u}_i)_j$ is an approximation of $u_i(x_j)$, we get

$$A^{-1}_{ij} = A^{-1}_{ji} = (\boldsymbol{u}_i)_j \approx u_i(x_j), \qquad i, j \in \{0, \dots, n-1\}. \tag{4.35}$$

We define $\widetilde{A}^{-1}$ to be the following pseudo-inverse of A:

$$\widetilde{A}^{-1}_{ij} = u_i(x_j) = \frac{\lambda}{\rho\cosh(\lambda)}\begin{cases} \sinh(\lambda(1-x_i))\cosh(\lambda x_j), & j \leqslant i \\ \sinh(\lambda(1-x_j))\cosh(\lambda x_i), & j > i \end{cases}, \qquad i, j \in \{0, \dots, n-1\}. \tag{4.36}$$

Using one of the product identities for hyperbolic trigonometric functions, we can expand $\widetilde{A}^{-1}_{ij}$ to

$$\widetilde{A}^{-1}_{ij} = \frac{\lambda}{2\rho\cosh(\lambda)}\begin{cases} \sinh(\lambda(1-x_i+x_j)) + \sinh(\lambda(1-x_i-x_j)), & j \leqslant i \\ \sinh(\lambda(1+x_i-x_j)) + \sinh(\lambda(1-x_i-x_j)), & j > i \end{cases}. \tag{4.37}$$

Lemma 7. *Let $\mathcal{T}_h$ be a uniform grid such as in* (4.30)*, and let α equal one of the grid nodes: $\alpha = x_{i_\alpha}$. Assume that u_h is the finite element solution satisfying* (4.20)*, and that u is the exact solution given by* (4.15) *for $\alpha = x_{i_\alpha}$. Then for each $i \in \{0, \dots, n-1\}$ we have*

$$|u(x_i) - u_h(x_i)| \leqslant ch^{3/2}$$

for some $c > 0$ independent of h.

[2]Here we use the slightly unconventional notation $\boldsymbol{e}_0 = (1, 0, \dots, 0)^\top$, $\boldsymbol{e}_1 = (0, 1, 0, \dots, 0)^\top$, etc.

Proof. Let $T_i = (x_i, x_{i+1})$ and define the error function $e_i(x) = |u(x) - u_h(x)|$ on T_i. We claim that $e_i \in C^2(T_i)$ with second derivative $e_i''(x) = |u_i''(x)|$, where $u_i = u|_{T_i}$. Indeed, note that u satisfies the differential equation

$$\rho u - \theta u'' = 0$$

on T_i. Thus we find

$$u'' = \frac{\rho}{\theta} u.$$

From expression (4.15) it can be deduced that u is non-negative and C^2 on each open interval not containing α. Therefore, u_i'' is non-negative and C^2 on T_i. Because $u_h''(x) = 0$, we get $e_i''(x) = |u_i''(x)|$. It follows that e_i satisfies the conditions of the trapezoid rule for approximating integrals. See for example [18, Theorem 5.3.3]. We get:

$$\begin{aligned}
e_i(x_i) &\leqslant e_i(x_i) + e_i(x_{i+1}) \\
&= \frac{2}{h} \left| \frac{h}{2} e_i(x_i) + e_i(x_{i+1}) \right| \\
&\leqslant \frac{2}{h} \left(\int_{T_i} e_i \, \mathrm{d}x + \gamma_i h^3 \right) \\
&\leqslant \frac{2}{h} \left(\sqrt{h} \sqrt{\int_{T_i} e_i^2 \, \mathrm{d}x} + \gamma_i h^3 \right) \\
&\leqslant \frac{2}{h} \left(\sqrt{h} \|e\|_{L^2(0,1)} + \gamma_i h^3 \right) \\
&\leqslant \frac{2}{h} \left(c_1 h^2 \sqrt{h} + \gamma_i h^3 \right) \\
&= c_2 h^{3/2} + c_3 h^2.
\end{aligned}$$

Here we have used the Cauchy-Schwarz inequality and Theorem 8. □

Although Lemma 7 shows that the pointwise error has order $\mathcal{O}(h^{3/2})$, we numerically verify that the pointwise error is actually of order $\mathcal{O}(h^2)$. See for example Figure 4.2, where we have plotted

$$\log_2 \left(\frac{E_{2n}}{E_n} \right), \tag{4.38}$$

where E_n is the maximum pointwise error between the exact solution and finite element solution on a uniform grid containing n elements.

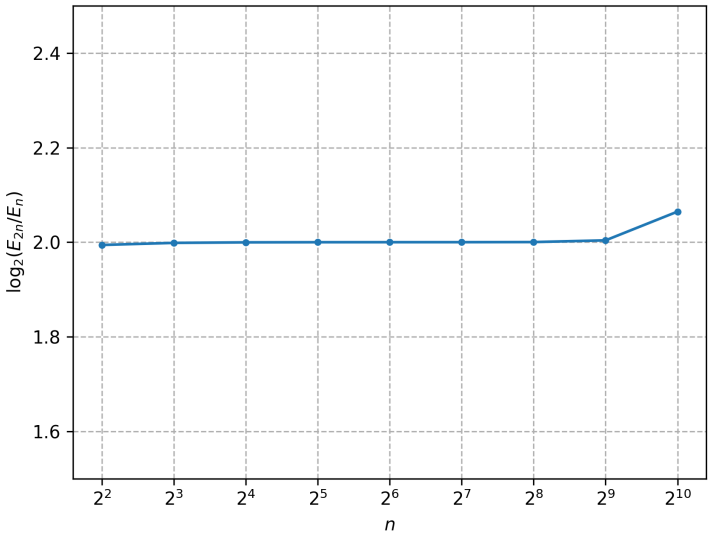


Figure 4.2: Numerical calculation of the order of the pointwise finite element error. We have set $\rho = \theta = 1$ and $i_\alpha = n/2$, corresponding to $\alpha = 0.5$.

We have not been able to prove this pointwise error bound, and therefore continue this section with the result of Lemma 7 in mind.

Theorem 9. *For each $i, j \in \{0, \dots, n-1\}$:*

$$\widetilde{A}_{ij}^{-1} - A_{ij}^{-1} = \mathcal{O}(h^{3/2}).$$

Proof. Assume $A\boldsymbol{u}_i = \boldsymbol{e}_i$ and $\widetilde{A}\widetilde{\boldsymbol{u}}_i = \boldsymbol{e}_i$, where $\widetilde{A} = (\widetilde{A}^{-1})^{-1}$. By the definition and symmetry of $\widetilde{A}^{-1}$, we have

$$(\widetilde{\boldsymbol{u}}_i)_j = \left(\widetilde{A}^{-1}\boldsymbol{e}_i\right)_j = \widetilde{A}_{ji}^{-1} = \widetilde{A}_{ij}^{-1} = u_i(x_j).$$

It then follows that

$$\widetilde{A}_{ij}^{-1} - A_{ij}^{-1} = \widetilde{A}_{ji}^{-1} - A_{ji}^{-1} = \left(\widetilde{A}^{-1}\boldsymbol{e}_i\right)_j - \left(A^{-1}\boldsymbol{e}_i\right)_j = u_i(x_j) - (\boldsymbol{u}_i)_j = \mathcal{O}(h^{3/2})$$

In the last step the result from Lemma 7 is applied. □

We will see later that it is sufficient to have an element-wise approximation of A^{-1}.

4.4 Determining the Optimal Tuning Parameter

Recall that our aim is to find an approximation of the matrix $C = DA^{-1}D^\top$. Using the results from the previous section, we can now approximate C. We first express its entries in terms of those of A^{-1}.

Lemma 8. *Let D be the finite element divergence matrix in the setting of Section 4.2. Then it is given by:*

$$D = \frac{1}{2}\begin{pmatrix} -1 & 1 & & & \\ -1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 0 & 1 \\ & & & -1 & 0 \end{pmatrix}, \tag{4.39}$$

where D is an $n \times n$ matrix, and we again use the index set $\{0, \dots, n-1\}$. Define the approximation $\widetilde{C} = D\widetilde{A}^{-1}D^\top$ to C. Then for $i, j \in \{1, \dots, n-2\}$:

$$\widetilde{C}_{ij} = \frac{1}{4}\left[\widetilde{A}_{i+1,j+1}^{-1} - \widetilde{A}_{i-1,j+1}^{-1} - \widetilde{A}_{i+1,j-1}^{-1} + \widetilde{A}_{i-1,j-1}^{-1}\right]. \tag{4.40}$$

Proof. The proof follows from combining the following two identities that hold for any $n \times n$ matrix B:

$$(DB)_{ij} = \frac{1}{2}\begin{cases} B_{1,j} - B_{0,j}, & i = 0 \\ B_{i+1,j} - B_{i-1,j}, & 1 \leqslant i \leqslant n-2 \\ -B_{n-2,j}, & i = n-1 \end{cases}, \tag{4.41}$$

and

$$(BD^\top)_{ij} = \frac{1}{2}\begin{cases} B_{i,1} - B_{i,0}, & j = 0 \\ B_{i,j+1} - B_{i,j-1}, & 1 \leqslant j \leqslant n-2 \\ -B_{i,n-2}, & j = n-1 \end{cases}. \tag{4.42}$$

Thus, for $i, j \in \{1, \dots, n-2\}$ we find

$$\begin{aligned} \left(D\widetilde{A}^{-1}D^\top\right)_{ij} &= \frac{1}{2}\left[\left(D\widetilde{A}^{-1}\right)_{i,j+1} - \left(D\widetilde{A}^{-1}\right)_{i,j-1}\right] \\ &= \frac{1}{2}\left[\frac{1}{2}\left[\widetilde{A}_{i+1,j+1}^{-1} - \widetilde{A}_{i-1,j+1}^{-1}\right] - \frac{1}{2}\left[\widetilde{A}_{i+1,j-1}^{-1} - \widetilde{A}_{i-1,j-1}^{-1}\right]\right] \\ &= \frac{1}{4}\left[\widetilde{A}_{i+1,j+1}^{-1} - \widetilde{A}_{i-1,j+1}^{-1} - \widetilde{A}_{i+1,j-1}^{-1} + \widetilde{A}_{i-1,j-1}^{-1}\right] \end{aligned}$$

□

Using Lemma 8 and equation (4.37), the entries of $\widetilde{C}$ can be computed exactly.

Theorem 10. *Let $\widetilde{C}$ again be defined by $\widetilde{C} = D\widetilde{A}^{-1}D^\top$, and let $i \in \{1, \dots, n-2\}$, then:*

$$\widetilde{C}_{ii} = \frac{\lambda}{4\rho\cosh(\lambda)}\Big[2h\lambda\cosh(\lambda(1-h)) + 2h^2\lambda^2\sinh(\lambda(1-2x_i))\Big] + \mathcal{O}(h^3). \tag{4.43}$$

$$\widetilde{C}_{i,i-1} = \frac{\lambda}{4\rho\cosh(\lambda)}\left[h\lambda\cosh(\lambda(1-2h)) + 2h^2\lambda^2\sinh(\lambda(1-2x_i+h))\right] + \mathcal{O}(h^3). \tag{4.44}$$

$$\widetilde{C}_{i,i+1} = \frac{\lambda}{4\rho\cosh(\lambda)}\left[h\lambda\cosh(\lambda(1-2h)) + 2h^2\lambda^2\sinh(\lambda(1-2x_i-h))\right] + \mathcal{O}(h^3) \tag{4.45}$$

Futhermore, for $1 \leqslant j \leqslant i-2$ we have:

$$\widetilde{C}_{ij} = \frac{h^2\lambda^3}{2\rho\cosh(\lambda)}\Big[\sinh\left(\lambda(1-2x_i+\delta h)\right) - \sinh\left(\lambda(1-\delta h)\right)\Big] + \mathcal{O}(h^4), \qquad \delta = i-j. \tag{4.46}$$

Likewise, for $i+2 \leqslant j \leqslant n-2$ we have:

$$\widetilde{C}_{ij} = \frac{h^2\lambda^3}{2\rho\cosh(\lambda)}\Big[\sinh\left(\lambda(1-2x_i-\delta h)\right) - \sinh\left(\lambda(1-\delta h)\right)\Big] + \mathcal{O}(h^4), \qquad \delta = j-i. \tag{4.47}$$

Proof. To avoid unnecessary repetition, we will only derive the expression for $\widetilde{C}_{ii}$. The other entries can be proved using a similar trick. Using (4.37) and Lemma 8, we find

$$\widetilde{C}_{ii} = \frac{\lambda}{8\rho\cosh(\lambda)}\Big[2(\sinh(\lambda) - \sinh(\lambda(1-2h))) \\ + \sinh(\lambda(1-2x_{i-1})) - 2\sinh(\lambda(1-2x_i)) + \sinh(\lambda(1-2x_{i+1}))\Big].$$

The first two terms between brackets look like a finite difference approximation to a first derivative. Similarly, the last three terms look like a finite difference approximation of a second derivative. Define

$$\psi(x) = \sinh(\lambda(1-x)),$$

then

$$\psi'(x) = -\lambda\cosh(\lambda(1-x)), \quad \text{and} \quad \psi''(x) = \lambda^2\sinh(\lambda(1-x)).$$

We use the following finite difference approximations:

$$\psi'(x) = \frac{\psi(x+h) - \psi(x-h)}{2h} + \mathcal{O}(h^2), \quad \text{and} \quad \psi''(x) = \frac{\psi(x+h) - 2\psi(x) + \psi(x-h)}{h^2} + \mathcal{O}(h^2).$$

From which it follows that

$$\begin{aligned}\sinh(\lambda) - \sinh(\lambda(1-2h)) &= \psi(0) - \psi(2h) \\ &= -2h\psi'(h) + \mathcal{O}(h^3) \\ &= 2h\lambda\cosh(\lambda(1-h)) + \mathcal{O}(h^3).\end{aligned}$$

Moreover:

$$\begin{aligned}\sinh(\lambda(1-2x_{i-1})) - 2\sinh(\lambda(1-2x_i)) + \sinh(\lambda(1-2x_{i+1})) &= \psi(2x_{i-1}) - 2\psi(2x_i) + \psi(2x_{i+1}) \\ &= \psi(2x_i - 2h) - 2\psi(2x_i) + \psi(2x_i + 2h) \\ &= 4h^2\psi''(x) + \mathcal{O}(h^4) \\ &= 4h^2\lambda^2\sinh(\lambda(1-2x_i)) + \mathcal{O}(h^4).\end{aligned}$$

Thus we find

$$\widetilde{C}_{ii} = \frac{\lambda}{4\rho\cosh(\lambda)}\Big[2h\lambda\cosh(\lambda(1-h)) + 2h^2\lambda^2\sinh(\lambda(1-2x_i))\Big] + \mathcal{O}(h^3)$$

Which completes the proof for $\widetilde{C}_{ii}$. To prove the identities for $\widetilde{C}_{i,i-1}$ and $\widetilde{C}_{i,i+1}$ one must also recognize finite difference quotients for a first and second derivative of ψ. To compute the remaining identities (4.46) and (4.47), two second derivative quotients should be used. □

Theorem 10 only considers the interior of $\widetilde{C}$ and not the first and last rows or columns. Note that we can expand Lemma 8 to also include $i, j \in \{0, n-1\}$ using the expressions (4.41) and (4.42). Because this leads to a large number of distinct cases, we have omitted these derivations from this report. It has been verified that expression such as the ones in Theorem 10 also hold for the entries in the first and last rows or columns of $\widetilde{C}$. Numerical experiments show that the entries become more accurate as they are further away from the diagonal.

From Theorem 9 we know that the entries of $\widetilde{A}^{-1}$ are accurate up to order $\mathcal{O}(h^{3/2})$. Thus, the order $\mathcal{O}(h^2)$ and higher terms in the entries of $\widetilde{C}$ will not contribute to a more accurate result. Let Q be the order h part of $\widetilde{C}$. By dividing the expressions in Theorem 10 by h and taking the limit $h \to 0$, we obtain for $i \in \{1, \ldots, n-2\}$:

$$(Q\boldsymbol{p})_i = \frac{h\lambda^2}{4\rho}\Big(p_{i-1} + 2p_i + p_{i+1}\Big) = \frac{h}{4\theta}\Big(p_{i-1} + 2p_i + p_{i+1}\Big). \tag{4.48}$$

And, as a direct result of Theorem 9:

$$(Q\boldsymbol{p})_i = (DA^{-1}D^\top \boldsymbol{p})_i + \mathcal{O}(h^{3/2}). \tag{4.49}$$

Consider again the coefficient matrix in the left-hand side of (4.11). Let $i \in \{1, \ldots, n-2\}$, then:

$$\begin{aligned}[(kL_p + \Delta t C)\boldsymbol{p}]_i &= [(kL_p + \Delta t Q)\boldsymbol{p}]_i + \mathcal{O}(h^2) \\ &= \left(\frac{h\Delta t}{4\theta} - \frac{k}{h}\right) p_{i-1} + \left(\frac{h\Delta t}{2\theta} + \frac{2k}{h}\right) p_i + \left(\frac{h\Delta t}{4\theta} - \frac{k}{h}\right) p_{i+1} + \mathcal{O}(h^{3/2})\end{aligned} \tag{4.50}$$

Stabilization is achieved by replacing kL_p with $(k+\beta)L_p$, we then get:

$$\begin{aligned}&[((k+\beta)L_p + \Delta t C)\boldsymbol{p}]_i = \\ &\qquad \left(\frac{h\Delta t}{4\theta} - \frac{k+\beta}{h}\right) p_{i-1} + \left(\frac{h\Delta t}{2\theta} + \frac{2(k+\beta)}{h}\right) p_i + \left(\frac{h\Delta t}{4\theta} - \frac{k+\beta}{h}\right) p_{i+1} + \mathcal{O}(h^{3/2})\end{aligned} \tag{4.51}$$

Let

$$\beta^* = \frac{h^2\Delta t}{4\theta} = \frac{h^2}{4(\mu_v + \lambda_v + \Delta t(\mu_\varepsilon + \lambda_\varepsilon))}. \tag{4.52}$$

Then β^* is the smallest value of β such that $(k+\beta)L_p + \Delta t Q$ is an M-matrix:

$$[((k+\beta^*)L_p + \Delta t C)\boldsymbol{p}]_i = -\frac{k}{h}p_{i-1} + \left(\frac{h\Delta t}{\theta} + \frac{2k}{h}\right) p_i - \frac{k}{h}p_{i+1} + \mathcal{O}(h^{3/2}) \tag{4.53}$$

Note that from (4.48) it follows that ρ does not affect the order h part of $\widetilde{C}$. Numerical experiments have confirmed that increasing or decreasing ρ has little to no effect on the occurance of non-physical oscillations.

If we assume that the two-sided stabilization of (4.3) is used, and β is set to β^*, then the pertubation in the corresponding differential equation (4.6) is - at least - of order $\mathcal{O}(\Delta t^2 h^{3/2})$. If the phenomenon seen in Figure 4.2 is taken into account, the order of the pertubation can be further sharpened to $\mathcal{O}(\Delta t^2 h^2)$.

4.5 Morpho-Poroelasticity

Combining the morpho- and poroelastic systems from Section 2.4 and 3.4 respectively yields the following system:

$$\rho\left(\frac{D\boldsymbol{v}}{Dt} + (\nabla \cdot \boldsymbol{v})\boldsymbol{v}\right) - \nabla \cdot \boldsymbol{\sigma}(\boldsymbol{v}, \boldsymbol{\varepsilon}, p) = \boldsymbol{g}, \tag{4.54}$$

$$\frac{D\boldsymbol{\varepsilon}}{Dt} + \boldsymbol{\varepsilon}\mathrm{skw}(\nabla \boldsymbol{v}) - \mathrm{skw}(\nabla \boldsymbol{v})\boldsymbol{\varepsilon} + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\,\mathrm{sym}(\nabla \boldsymbol{v}) = -\boldsymbol{G}, \tag{4.55}$$

$$\nabla \cdot \boldsymbol{v} - \nabla \cdot (k\nabla p) = f. \tag{4.56}$$

Where $\boldsymbol{G}$ is the growth tensor. The above equations must be satisfied on the moving domain $\Omega(t)$. The stress tensor $\boldsymbol{\sigma}$ in this framework is given by:

$$\boldsymbol{\sigma}(\boldsymbol{v}, \boldsymbol{\varepsilon}, p) = \boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) - p\boldsymbol{I}, \tag{4.57}$$

where

$$\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) = \mu_\varepsilon \boldsymbol{\varepsilon} + \lambda_\varepsilon \mathrm{tr}(\boldsymbol{\varepsilon})\boldsymbol{I},$$
$$\boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) = \frac{1}{2}\mu_v \left(\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^\top\right) + \lambda_v (\nabla \cdot \boldsymbol{v})\boldsymbol{I}.$$

The same boundary conditions as for the visco- poroelastic system are imposed:

$$\boldsymbol{v} = \boldsymbol{0}, \quad k\nabla p \cdot \boldsymbol{n} = 0, \qquad \text{on } \Gamma_1(t), \tag{4.58}$$
$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\tau}, \quad p = 0, \qquad \text{on } \Gamma_2(t). \tag{4.59}$$

We assume that $\Gamma(t)$ is the piecewise smooth boundary of $\Omega(t)$, such that $\Gamma(t) = \Gamma_1(t) \cup \Gamma_2(t)$ and $\Gamma_1(t) \cap \Gamma_2(t) = \varnothing$ for all $t > 0$.

4.5.1 Relation to Morphoelastic System

The morpho-poroelastic system is closely related to the morphoelastic system. We show that as k approaches infinity the solutions for $\boldsymbol{v}$ and $\boldsymbol{\varepsilon}$ of the morpho-poroelastic system approach the corresponding solutions to the morphoelastic system. Intuitively, increasing k means either the permeability of the porous medium increases, or the dynamic viscosity of the poro-fluid decreases. In both cases the fluid is able to move more freely throughout the medium as it deforms, having less impact on the domain deformation. This is formalised in the following theorem.

Theorem 11. *Let $\boldsymbol{v}_k$, $\boldsymbol{\varepsilon}_k$ and p_k be the solutions to the morpho-poroelastic system* (4.54)-(4.56) *with boundary conditions*

$$\boldsymbol{v} = \boldsymbol{0}, \quad k\nabla p \cdot \boldsymbol{n} = J, \qquad \textit{on } \Gamma_1(t), \tag{4.60}$$
$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\tau}, \quad p = p_0, \qquad \textit{on } \Gamma_2(t), \tag{4.61}$$

where J is bounded and p_0 is constant. Let $\boldsymbol{v}$ and $\boldsymbol{\varepsilon}$ be the solutions to the morphoelastic system (4.58)-(4.59) *with boundary conditions*

$$\boldsymbol{v} = \boldsymbol{0}, \qquad \textit{on } \Gamma_1(t), \tag{4.62}$$
$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\tau}, \qquad \textit{on } \Gamma_2(t). \tag{4.63}$$

Then $\boldsymbol{v}_k \to \boldsymbol{v}$ and $\boldsymbol{\varepsilon}_k \to \boldsymbol{\varepsilon}$ as $k \to \infty$.

Proof. We give the outline of the proof. Some properties of the solutions are assumed without them being proved. In a future work, these technicalities should be addressed. Note that in the limit $k \to \infty$, equation (4.56) turns into

$$\nabla^2 p = 0,$$

with boundary conditions

$$\nabla p \cdot \boldsymbol{n} = 0, \quad \text{on } \Gamma_1(t), \qquad p = p_0, \quad \text{on } \Gamma_2(t).$$

This holds provided $\nabla \cdot \boldsymbol{v}$ remains bounded on $\Omega(t)$ for all t. Note that since $\boldsymbol{v}$ has been eliminated from (4.56), the equation has been decoupled from (4.54) and (4.55) but not vice versa, since p still occurs in (4.54). The resulting boundary value problem for p has the unique constant solution $p(\boldsymbol{x}, t) = p_0$ on $\Omega(t)$. If we substitute this solution into the stress tensor (4.57) and take the divergence, we obtain:

$$\begin{aligned}\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{v}, \boldsymbol{\varepsilon}, p_0) &= \nabla \cdot (\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v})) - \nabla p_0 \\ &= \nabla \cdot (\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}))\,,\end{aligned}$$

because ∇p_0 vanishes. Thus, we find that $\boldsymbol{v}$ and $\boldsymbol{\varepsilon}$ satisfy the morphoelastic system (4.58)-(4.59). □

4.5.2 Derivation of System of Equations

We present the weak forms corresponding to equations (4.54)-(4.56). Their complete derivations will not be discussed, as most of them have already been shown a number of times throughout this work. The weak form of (4.54) is similar to (3.91):

$$\rho \frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \boldsymbol{v} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega + \int_{\Omega(t)} (\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v})) : \nabla \boldsymbol{\varphi} \,\mathrm{d}\Omega - \int_{\Omega(t)} p(\nabla \cdot \boldsymbol{\varphi}) \,\mathrm{d}\Omega$$
$$= \int_{\Omega(t)} \boldsymbol{g} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Omega + \int_{\Gamma_2(t)} \boldsymbol{\tau} \cdot \boldsymbol{\varphi} \,\mathrm{d}\Gamma. \tag{4.64}$$

Where $\boldsymbol{\varphi}$ is a vector-valued Lagriangian test field that vanishes on $\Gamma_1(t)$. We show the derivation of the term involving the stress tensor. Using Green's formula, Gauss' divergence theorem and the boundary conditions, we get:

$$\begin{aligned}
-\int_{\Omega(t)} (\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega &= \int_{\Omega(t)} \boldsymbol{\sigma} : \nabla \boldsymbol{\varphi} - \nabla \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{\varphi}) \, \mathrm{d}\Omega \\
&= \int_{\Omega(t)} (\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) - p\boldsymbol{I}) : \nabla \boldsymbol{\varphi} \, \mathrm{d}\Omega - \int_{\Gamma(t)} (\boldsymbol{n} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{\varphi} \, \mathrm{d}\Gamma \\
&= \int_{\Omega(t)} (\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v})) : \nabla \varphi \, \mathrm{d}\Omega - \int_{\Omega(t)} p(\nabla \cdot \boldsymbol{\varphi}) \, \mathrm{d}\Omega - \int_{\Gamma_2(t)} \boldsymbol{\tau} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Gamma.
\end{aligned}$$

The weak form of (4.55) is derived in (2.106), it is given by:

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \boldsymbol{\varepsilon} : \boldsymbol{\zeta} \, \mathrm{d}\Omega \\
&\qquad + \int_{\Omega(t)} \Big(\boldsymbol{\varepsilon} \mathrm{skw}(\nabla \boldsymbol{v}) - \mathrm{skw}(\nabla \boldsymbol{v}) \boldsymbol{\varepsilon} + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1) \mathrm{sym}(\nabla \boldsymbol{v}) - (\nabla \cdot \boldsymbol{v}) \boldsymbol{\varepsilon} \Big) : \boldsymbol{\zeta} \, \mathrm{d}\Omega \\
&\qquad\qquad = -\int_{\Omega(t)} \boldsymbol{G} : \boldsymbol{\zeta} \, \mathrm{d}\Omega. \quad (4.65)
\end{aligned}$$

Here $\boldsymbol{\zeta}$ is a tensor-valued Lagrangian test field. Finally, the weak form of equation (4.56) is given by

$$\int_{\Omega(t)} (\nabla \cdot \boldsymbol{v}) \psi \, \mathrm{d}\Omega + \int_{\Omega(t)} k \nabla p \cdot \nabla \psi \, \mathrm{d}\Omega = \int_{\Omega(t)} f \psi \, \mathrm{d}\Omega \tag{4.66}$$

Where ψ is a scalar-valued Lagrangian test function. Weak forms (4.64), (4.65) and (4.66) are converted to a semi-discretised algebraic system by computing the Galerkin equations. To this end, the approximate domain $\Omega_h(t)$ is triangulated by $\mathcal{T}_h(t)$. The vectors $\boldsymbol{v}, \boldsymbol{\varepsilon}$ and $\boldsymbol{p}$ contain the values of their corresponding unknown in the grid points. By following the usual steps we arrive at the following set of ordinary differential equations:

$$\rho \frac{\mathrm{d}}{\mathrm{d}t} (M_v \boldsymbol{v}) + S_{\mathrm{vis}} \boldsymbol{v} + S_{\mathrm{el}} \boldsymbol{\varepsilon} - D^\top \boldsymbol{p} = \boldsymbol{g} \tag{4.67}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} (M_\varepsilon \boldsymbol{\varepsilon}) - B \boldsymbol{v} + \boldsymbol{N}(\boldsymbol{v}, \boldsymbol{\varepsilon}) = -\boldsymbol{G} \tag{4.68}$$

$$D\boldsymbol{v} + kL\boldsymbol{p} = \boldsymbol{f} \tag{4.69}$$

Here $\boldsymbol{v} = (\boldsymbol{v}^1, \boldsymbol{v}^2)^\top$, $\boldsymbol{\varepsilon} = (\boldsymbol{\varepsilon}^{11}, \boldsymbol{\varepsilon}^{12}, \boldsymbol{\varepsilon}^{22})^\top$ and $\boldsymbol{p}$ are the vectors containing the unknowns, and $\boldsymbol{g} = (\boldsymbol{g}^1, \boldsymbol{g}^2)^\top$ and $\boldsymbol{G} = (\boldsymbol{G}^{11}, \boldsymbol{G}^{12}, \boldsymbol{G}^{22})^\top$ adn $\boldsymbol{f}$ are the right-hand side vectors. The symmetry of $\boldsymbol{\varepsilon}$ has been used to eliminate ε^{21} from the system. Moreover, the two mass matrices are given by:

$$M_v = \begin{pmatrix} M & 0 \\ 0 & M \end{pmatrix}, \qquad M_\varepsilon = \begin{pmatrix} M & 0 & 0 \\ 0 & M & 0 \\ 0 & 0 & M \end{pmatrix}, \tag{4.70}$$

where M is the mass matrix given in (3.96). Note that here we assume $\boldsymbol{v}$ and $\boldsymbol{\varepsilon}$ both use linear basis functions. Since the divergence matrix D is the same as in (3.62), it again consists of the two blocks $D = \begin{pmatrix} D^1 & D^2 \end{pmatrix}$. From (2.109) we see that the matrices B and S_{el} can be expressed in terms of D^1 and D^2:

$$B = \begin{pmatrix} D^1 & 0 \\ \frac{1}{2} D^2 & \frac{1}{2} D^1 \\ 0 & D^2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{2} \\ 0 & 0 \end{pmatrix} \otimes D^1 + \begin{pmatrix} 0 & 0 \\ \frac{1}{2} & 0 \\ 0 & 1 \end{pmatrix} \otimes D^2, \tag{4.71}$$

and

$$S_{\mathrm{el}}^\top = \begin{pmatrix} (\mu_\varepsilon + \lambda_\varepsilon) D^1 & \lambda_\varepsilon D^2 \\ \mu_\varepsilon D^2 & \mu_\varepsilon D^1 \\ \lambda_\varepsilon D^1 & (\mu_\varepsilon + \lambda_\varepsilon) D^2 \end{pmatrix} = \begin{pmatrix} \mu_\varepsilon + \lambda_\varepsilon & 0 \\ 0 & \mu_\varepsilon \\ \lambda_\varepsilon & 0 \end{pmatrix} \otimes D^1 + \begin{pmatrix} 0 & \lambda_\varepsilon \\ \mu_\varepsilon & 0 \\ 0 & \mu_\varepsilon + \lambda_\varepsilon \end{pmatrix} \otimes D^2. \tag{4.72}$$

Here $\otimes$ denotes the Kronecker product. Note that the elasticity matrix S_{el} now acts on $\boldsymbol{\varepsilon}$ instead of the displacement $\boldsymbol{u}$. The matrix S_{vis} is well-known viscous stifness matrix of which its elements are given by

(2.71)-(2.74). The non-linear operator $\boldsymbol{N}$ is defined in Section 2.4.2. We apply implicit Euler to system (4.67)-(4.69). Let a superscript $(\cdot)^m$ denote a time-level. Note that all discrete operators depend on the current grid, and therefore also get a time-level superscript. We obtain the fully discrete system:

$$(\rho M_v^m + \Delta t S_{\text{vis}}^m)\,\boldsymbol{v}^m + \Delta t S_{\text{el}}^m \boldsymbol{\varepsilon}^m - \Delta t (D^m)^\top \boldsymbol{p}^m = \rho M_v^{m-1}\boldsymbol{v}^{m-1} + \Delta t \boldsymbol{g}^m, \tag{4.73}$$

$$M_\varepsilon^m \boldsymbol{\varepsilon}^m - \Delta t B^m \boldsymbol{v}^m + \Delta t \boldsymbol{N}^m(\boldsymbol{v}^m, \boldsymbol{\varepsilon}^m) = M_\varepsilon^{m-1}\boldsymbol{\varepsilon}^{m-1} - \Delta t \boldsymbol{G}^m, \tag{4.74}$$

$$D^m \boldsymbol{v}^m + kL^m \boldsymbol{p}^m = \boldsymbol{f}^m. \tag{4.75}$$

Equivalently, in block-matrix notation:

$$\begin{pmatrix} \rho M_v^m + \Delta t S_{\text{vis}}^m & \Delta t S_{\text{el}}^m & -\Delta t (D^m)^\top \\ -\Delta t B^m & M_\varepsilon^m & 0 \\ D^m & 0 & kL^m \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^m \\ \boldsymbol{\varepsilon}^m \\ \boldsymbol{p}^m \end{pmatrix} + \Delta t \begin{pmatrix} 0 \\ \boldsymbol{N}^m(\boldsymbol{v}^m, \boldsymbol{\varepsilon}^m) \\ 0 \end{pmatrix} = \begin{pmatrix} \rho M_v^{m-1}\boldsymbol{v}^{m-1} \\ M_\varepsilon^{m-1}\boldsymbol{\varepsilon}^{m-1} \\ 0 \end{pmatrix} + \begin{pmatrix} \Delta t \boldsymbol{g}^m \\ -\Delta t \boldsymbol{G}^m \\ \boldsymbol{f}^m \end{pmatrix}. \tag{4.76}$$

Note that to solve system (4.76), the operators and right-hand side vectors at time-level m must be known. However, these depend on the grid at time-level m and thus on the solution of the system. As such, an iterative method is used to solve (4.76). Details regarding this method can be found in section 2.4.3.

4.5.3 Stabilization

As was the case in the previous poroelastic systems, equation (4.56) will cause non-physical oscillations in the finite element solution for small values of k. To find out which matrix is causing this behaviour, write system (4.76) in the following abbreviated form:

$$\begin{pmatrix} A & \Delta t S_{\text{el}} & -\Delta t D^\top \\ -\Delta t B & M_\varepsilon & 0 \\ D & 0 & kL \end{pmatrix} \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{\varepsilon} \\ \boldsymbol{p} \end{pmatrix} + \Delta t \begin{pmatrix} 0 \\ \boldsymbol{N}(\boldsymbol{v}, \boldsymbol{\varepsilon}) \\ 0 \end{pmatrix} = \begin{pmatrix} \boldsymbol{r}^1 \\ \boldsymbol{r}^2 \\ \boldsymbol{r}^3 \end{pmatrix}. \tag{4.77}$$

Here we have dropped the time-level superscripts and set $A = \rho M_v + \Delta t S_{\text{vis}}$. Solve the first equation for $\boldsymbol{v}$:

$$\boldsymbol{v} = \Delta t A^{-1} D^\top \boldsymbol{p} - \Delta t A^{-1} S_{\text{el}} \boldsymbol{\varepsilon} + A^{-1}\boldsymbol{r}^1. \tag{4.78}$$

Substitute the above result in the third equation to obtain:

$$\left(kL + \Delta t D A^{-1} D^\top\right) \boldsymbol{p} = \Delta t D A^{-1} S_{\text{el}} \boldsymbol{\varepsilon} - D A^{-1}\boldsymbol{r}^1 + \boldsymbol{r}^3. \tag{4.79}$$

The left-hand side of the above equation has the same structure as (4.11), but with a different matrix A. By repeating the derivation of Sections 4.2, 4.3 and 4.4 for the morpho-poroelastic system, we obtain the following optimal tuning parameter:

$$\beta^* = \frac{h^2}{4(\mu_v + \lambda_v)}. \tag{4.80}$$

This result is then extrapolated to the two-dimensional morpho-poroelastic system on a non-uniform grid. The complete stabilized version of system (4.76) is given by:

$$\begin{pmatrix} \rho M_v^m + \Delta t S_{\text{vis}}^m & \Delta t S_{\text{el}}^m & -\Delta t (D^m)^\top \\ -\Delta t B^m & M_\varepsilon^m & 0 \\ D^m & 0 & kL^m + C_s^m \end{pmatrix} \begin{pmatrix} \boldsymbol{v}^m \\ \boldsymbol{\varepsilon}^m \\ \boldsymbol{p}^m \end{pmatrix} + \Delta t \begin{pmatrix} 0 \\ \boldsymbol{N}^m(\boldsymbol{v}^m, \boldsymbol{\varepsilon}^m) \\ 0 \end{pmatrix} = \begin{pmatrix} \rho M_v^{m-1}\boldsymbol{v}^{m-1} \\ M_\varepsilon^{m-1}\boldsymbol{\varepsilon}^{m-1} \\ C_s^{m-1}\boldsymbol{p}^{m-1} \end{pmatrix} + \begin{pmatrix} \Delta t \boldsymbol{g}^m \\ -\Delta t \boldsymbol{G}^m \\ \boldsymbol{f}^m \end{pmatrix}, \tag{4.81}$$

where the stabilization matrix C_s is given by:

$$(C_s)_{ij}^m = \frac{1}{4(\mu_v + \lambda_v)} \sum_{T \in \mathcal{T}^m} h_T^2 \int_T \nabla\varphi_i \cdot \nabla\varphi_j \,\mathrm{d}\Omega. \tag{4.82}$$

Here $\mathcal{T}^m$ denotes the triangulation of the domain at time-level m.

4.6 Comparison Between Morphoelastic and Morpho-Poroelastic Models

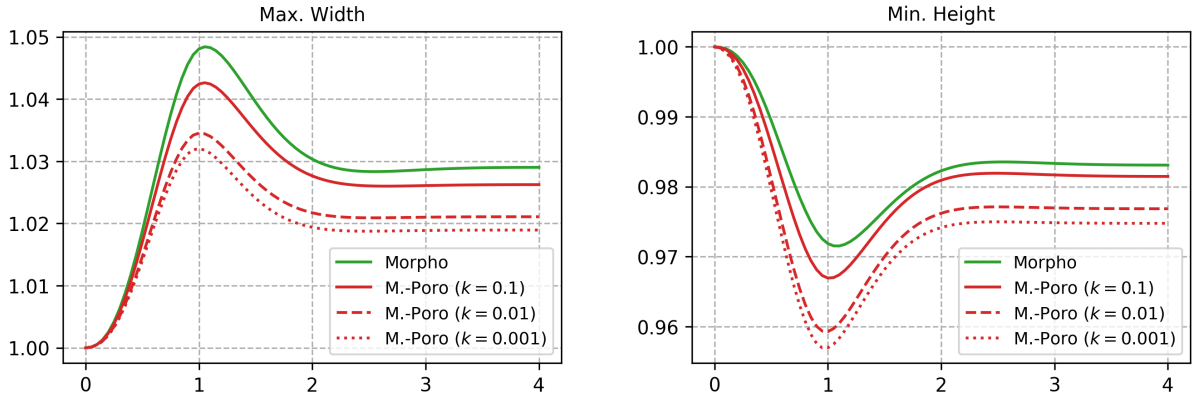
We compare the behaviour of the morpho-poroelastic model compared to the morphoelastic model from Section 2.4. To this end, a similar configuration as in Section 3.5 is used, where we compare the visco-poroelastic model to the viscoelastic model. Thus, the following body force is applied to the domain:

$$\boldsymbol{g}(\boldsymbol{x}, t) = \begin{pmatrix} 0.2\sin(\pi t)\mathbb{1}_{[0,1]}(t) \\ 0 \end{pmatrix}. \tag{4.83}$$

The initial domain is again the unit square. The mass source function f is set to zero, and the following mechanical parameters are used:

$$\mu_\varepsilon = \lambda_\varepsilon = 1, \quad \mu_v = \lambda_v = 0.5, \quad \rho = 0.5. \tag{4.84}$$

Furthermore, we set the growth parameter α equal to 1 for both models. We use the (initially) uniform finite element mesh from Figure 2.10 and a timestep of size $\Delta t = 0.05$. Figures 4.3a and 4.3b show the maximum domain width and mimimum domain height as functions of time, for various morphoelastic models. The red lines correspond to the morpho-poroelastic simulations for various values of k, whereas the green line corresponds to the morphoelastic model from Section 2.4.



(a) Maximum width of domain.

(b) Minimum height of domain.

Figure 4.3: Maximum domain width and mimimum domain height as functions of time for morphoelastic model and three morpho-poroelastic models.

Note that for smaller values of k, there is less extension is the x-direction (width), but more compression in the y-direction (height). It also seems that the viscoelastic 'overshoot' property is less pronounced for lower values of k. For higher values of k, we see that the solution to the morpho-poroelastic system approaches the solution to the morphoelastic system. This is in line with the theory from Section 4.5.1

5

Tumor Growth

In this chapter we apply the developed morpho-poroelastic model to tumor growth. To this end, the framework of [14] is used. We will first give a brief summary of this framework.

5.1 Mathematical Model

The model proposed in [14] is based on the idea that a tumor grows better in areas where there is a high concentration of oxygen. The presence of oxygen will therefore encourage tumor growth. The new tissue that is created will exert a force on its surroundings, which has to be incorporated in the force balance equation. Let η be the volume of new tissue created per unit volume of tissue. To model the force exerted by the new tissue on the medium, the following term is added the stress tensor:

$$-\left(\frac{\mu_\varepsilon}{3}+\lambda_\varepsilon\right)\eta\boldsymbol{I} \tag{5.1}$$

The parameters λ_ε and μ_ε are the familiar Lamé constants, such that the constant $\lambda_\varepsilon + \mu_\varepsilon/3$ equals the bulk modulus of the drained medium. Moreover, depending on whether the underlying model is purely elastic, viscoelastic or morphoelastic, the elastic stress tensor $\boldsymbol{\sigma}_{\text{el}}$ can be adjusted. In the latter two cases, an intertial term must be included. The change in time of η depends on the oxygen concentration c and the fraction of the medium occupied by cells, which is equal to the solid part of the medium Φ. Therefore, $1-\Phi$ is the fraction of space that is occupied by fluid. It is important to note that Φ is not a constant; it can change in time and space according to the following equation [14, Eq. 1]:

$$\frac{\partial\Phi}{\partial t}+\nabla\cdot(\Phi\boldsymbol{v})=S. \tag{5.2}$$

The function S is the cell creation source. It also acts as a source function for η in the sense that $\partial\eta/\partial t = S$. In [14] the following cell creation source function is proposed:

$$S=F(\overline{\sigma})\Phi(1-\Phi)N_{\text{G}}(c)-r_{\text{d}}\Phi, \tag{5.3}$$

where $N_{\text{G}}(c)$ is the Michaelis-Menten cell growth rate as a function of the nutrient concentration. It is given by:

$$N_{\text{G}}(c)=\frac{G_{\max}c}{c_G+c}. \tag{5.4}$$

The parameter $G_{\max}$ is the maximum cell growth rate when there is no shortage of oxygen, and c_G is the Michaelis constant, which is equal to the oxygen concentration for which the cell creation rate is half of $G_{\max}$. Looking back at (5.3), the parameter r_{d} is the death rate of tumor cells. The term r_{d} models the fact that tumor cells die in the absence of oxygen. The factor $F(\overline{\sigma})$ is a stress-dependent growth factor. In [14], F is chosen equal to $F(\overline{\sigma}) = 1-\beta\overline{\sigma}$. The parameter β is chosen to equal

$$\beta=\frac{3}{\sqrt[3]{2}}\left(\frac{1}{\mu_\varepsilon+3\lambda_\varepsilon}+\frac{1}{2\mu_\varepsilon}\right). \tag{5.5}$$

This seemingly arbitrary choice is derived in [14] by calculating the stress needed for a spherical cell to increase its radius. If we repeat this derivation for a circular cell in two dimensions, we find:

$$\beta=\sqrt{2}\left(\frac{1}{\mu_\varepsilon+3\lambda_\varepsilon}+\frac{1}{2\mu_\varepsilon}\right), \tag{5.6}$$

under the assumption that [14, Eq (12)] holds for circular cells. Note the factor $\sqrt{2} = 2/\sqrt{2}$ instead of $3/\sqrt[3]{2}$, which is the result of using area (squares) instead of volume (cubes). Moreover, $\overline{\sigma}$ is the average of the radial and circumferential stresses. In three dimensions, we would have:

$$\overline{\sigma} = \frac{\sigma_{rr} + \sigma_{\theta\theta} + \sigma_{\varphi\varphi}}{3}, \tag{5.7}$$

where θ and φ are respectively the azimuthal and polar angle. Using the fact that the trace of a matrix is invariant under a change of basis, we also have

$$\overline{\sigma} = \frac{\sigma_{xx} + \sigma_{yy} + \sigma_{zz}}{3}. \tag{5.8}$$

Similarly, in two dimensions:

$$\overline{\sigma} = \frac{\sigma_{xx} + \sigma_{yy}}{2}. \tag{5.9}$$

Note that $\overline{\sigma}$ depends on the elastic model that is used. For example:

- For the visco-poroelastic model from Section 3.4, we have

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{u}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) - p\boldsymbol{I} - \left(\frac{\mu_\varepsilon}{3} + \lambda_\varepsilon\right)\eta\boldsymbol{I}, \tag{5.10}$$

  where the viscous stress tensor is $\boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) = \mu_v \mathrm{sym}(\nabla\boldsymbol{v}) + \lambda_v(\nabla\cdot\boldsymbol{v})\boldsymbol{I}$. In this case

$$\overline{\sigma} = \left(\frac{\mu_\varepsilon}{2} + \lambda_\varepsilon\right)(\nabla\cdot\boldsymbol{u}) + \left(\frac{\mu_v}{2} + \lambda_v\right)(\nabla\cdot\boldsymbol{v}) - p - \left(\frac{\mu_\varepsilon}{3} + \lambda_\varepsilon\right)\eta. \tag{5.11}$$

- For the morpho-poroelastic model from Section 4.5, we would have

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) - p\boldsymbol{I} - \left(\frac{\mu_\varepsilon}{3} + \lambda_\varepsilon\right)\eta\boldsymbol{I}. \tag{5.12}$$

  The elasticity tensor is now given by $\boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) = \mu_\varepsilon\boldsymbol{\varepsilon} + \lambda_\varepsilon \mathrm{tr}(\boldsymbol{\varepsilon})\boldsymbol{I}$. We then get

$$\overline{\sigma} = \left(\frac{\mu_\varepsilon}{2} + \lambda_\varepsilon\right)\mathrm{tr}(\boldsymbol{\varepsilon}) + \left(\frac{\mu_v}{2} + \lambda_v\right)(\nabla\cdot\boldsymbol{v}) - p - \left(\frac{\mu_\varepsilon}{3} + \lambda_\varepsilon\right)\eta. \tag{5.13}$$

Finally, consider the effect of Φ on S. If Φ is equal to 0, then there are no tumor cells present that can multiply, hence no growth is possible. On the other hand, if Φ is equal to 1, the fluid can no longer reach the tumor to deliver oxygen. The transport of oxygen throughout the medium is modeled using a convection-diffusion equation. Convection is driven by the fluid velocity $\boldsymbol{v}_{\mathrm{F}}$, which is related to the Darcy flux as follows:

$$(1 - \Phi)\boldsymbol{v}_{\mathrm{F}} = -k\nabla p. \tag{5.14}$$

To model the absorption of oxygen into the tissue, a nutrient absorption function $N_{\mathrm{A}}(c)$ that acts as a sink is added to the equation. We assume that oxygen is the only nutrient that allows tumor cells to grow. Oxygen absorption also follows a Michaelis-Menten type equation:

$$N_{\mathrm{A}}(c) = \frac{A_{\max} c}{c_A + c}, \tag{5.15}$$

where $A_{\max}$ is the maximum absorption rate, and c_A is the concentration for which the absorption rate is half of its maximum. By also incorporating the deformation of the medium, and a diffusive term, we get the following transport equation for c:

$$\frac{\partial c}{\partial t} + \nabla\cdot(\boldsymbol{v}c) - \nabla\cdot(k\nabla p c) = \lambda_N\nabla^2 c - \Phi N_{\mathrm{A}}(c) \tag{5.16}$$

The parameter λ_N is the oxygen (nutrient) diffusion constant. In the following sections, we derive a finite element model to solve convection-diffusion equations such as (5.16). First, a simpler equation is solved on a fixed grid and constant fluid velocity. To avoid oscillations in convection dominated systems, the Streamline Upwind Petrvov-Galerkin (SUPG) method can be used. We have found this to be unnecessary in our numerical experiments, because the convection remained small compared to the diffusion, especially near the domain boundaries. In Sections 5.2.1 and 5.2.2 we consider two 'zero-dimensional' systems, to investigate the behaviour of nutrient absorption and its relation to tissue growth. In subsequent sections, we build up to the complete model that combines a (visco/morpho-) elastic model with oxygen transport and cell creation.

5.1.1 Complete Model Equations

In this section we combine the mechanical model (that is, morpho-/visco-/poroelasticity), with the biochemical tumor growth model. On the one hand, we use the model from Section 4.5 with and additional term that models the stress exerted by tumor cells on the tissue. The mechanical model equations are given by:

$$\rho\left(\frac{D\boldsymbol{v}}{Dt} + (\nabla\cdot\boldsymbol{v})\boldsymbol{v}\right) - \nabla\cdot\boldsymbol{\sigma}(\boldsymbol{v},\boldsymbol{\varepsilon},p,\eta) = \boldsymbol{g}, \tag{5.17}$$

$$\frac{D\boldsymbol{\varepsilon}}{Dt} + \boldsymbol{\varepsilon}\mathrm{skw}(\nabla\boldsymbol{v}) - \mathrm{skw}(\nabla\boldsymbol{v})\boldsymbol{\varepsilon} + (\mathrm{tr}(\boldsymbol{\varepsilon}) - 1)\,\mathrm{sym}(\nabla\boldsymbol{v}) = -\boldsymbol{G}, \tag{5.18}$$

$$\nabla\cdot\boldsymbol{v} - \nabla\cdot(k\nabla p) = f. \tag{5.19}$$

The stress tensor is given by:

$$\boldsymbol{\sigma}(\boldsymbol{v},\boldsymbol{\varepsilon},p,\eta) = \boldsymbol{\sigma}_{\mathrm{el}}(\boldsymbol{\varepsilon}) + \boldsymbol{\sigma}_{\mathrm{vis}}(\boldsymbol{v}) - p\boldsymbol{I} - \left(\frac{\mu_\varepsilon}{3} + \lambda_\varepsilon\right)\eta\boldsymbol{I} \tag{5.20}$$

The elastic and viscous stress tensor are given by (2.89) and (2.39) respectively. We set the growth tensor equal to $\boldsymbol{G} = \alpha\boldsymbol{\varepsilon}$ for $\alpha \geqslant 0$, such that we can model permanent deformations. On the other hand, the biological model from [14] is used to model oxygen transport and tumor growth.

$$\frac{Dc}{Dt} + (\nabla\cdot\boldsymbol{v})c - \nabla\cdot(kc\nabla p + \lambda_N\nabla c) = -\Phi N_{\mathrm{A}}(c), \tag{5.21}$$

$$\frac{D\Phi}{Dt} + (\nabla\cdot\boldsymbol{v})\Phi = F(\overline{\sigma})\Phi(1-\Phi)N_{\mathrm{G}}(c) - r_{\mathrm{d}}\Phi. \tag{5.22}$$

$$\frac{\partial\eta}{\partial t} = F(\overline{\sigma})\Phi(1-\Phi)N_{\mathrm{G}}(c) - r_{\mathrm{d}}\Phi. \tag{5.23}$$

Thus, we have six equations and the six corresponding unknowns $\boldsymbol{v}, \boldsymbol{\varepsilon}, p, c, \Phi$ and η. Note however that there are nine scalar unknowns; $\boldsymbol{v}$ has components v^1, v^2, and $\boldsymbol{\varepsilon}$ has the (unique) components $\varepsilon^{11}, \varepsilon^{12}, \varepsilon^{22}$. We use linear basis functions with respect to a triangular mesh for all unknowns. The finite element discretization of equations (5.17)-(5.19) can be found in Section 4.5. The discretization of the term involving $\nabla\eta$ in (5.17) is done similarly to the ∇p term. Discretization of (5.21) and (5.22) is discussed in Section 5.3. Equation (5.23) is discussed in Section 5.3.3.

It is important to note that the mechanical and biological models are two-way coupled. The mechanical model produces the solid velocity $\boldsymbol{v}$ and the pressure p, which are necessary to compute the oxygen concentration. Moreover, the average stress $\overline{\sigma}$ is needed to calculate the volume fraction Φ and new tissue volume η. On the other hand, the biological model also affects the mechanics through the $\nabla\eta$ term in the force balance equation.

Due to the interconnectedness of the equations, we choose a monolithic approach over some kind of staggered approach. A possible staggered approach would be to first solve the mechanical model, and use its output to solve the biological model. In the next step the output of the biological model is used to solve the mechanical model again. The advantage of such a method is that two smaller systems are solved instead of one large system. However, the coefficient matrix of the mechanical model is already $6n \times 6n$ where n is the number of meshpoints. The biological model is much smaller in comparison; only $3n \times 3n$. Thus, a monolithic approach will amount to solving a $9n \times 9n$ system. Additionally, a staggered approach is often very useful for converting a non-linear system into two linear systems. If we were to use a staggered method then the variable p in equation (5.21) would be known, effectively making this term linear. However, both the mechanical and biological systems are inherently non-linear, a staggered approach will not help in this sense.

5.1.2 Testing Configurations

We present the different testing configurations used to test the model. Note that due to the nature of equation (5.18) and (5.22), the unknowns $\boldsymbol{\varepsilon}, \Phi$ and η do not need boundary conditions.

(i) The first configuration is a square with one rigid edge. We set

$$\Omega(0) = (0,1)^2, \quad \Gamma_1(0) = \{(x,y) : x = 0\}, \quad \Gamma_2(0) = \partial\Omega(0) \setminus \Gamma_1(0).$$

On $\Gamma_1(t)$, we impose either the boundary conditions:

$$\boldsymbol{v} = \boldsymbol{0}, \quad \boldsymbol{n} \cdot k\nabla p = 0, \quad c = c_b. \tag{5.24}$$

or

$$\boldsymbol{v} = \boldsymbol{0}, \quad \boldsymbol{n} \cdot k\nabla p = 0, \quad \boldsymbol{n} \cdot (ck\nabla p + \lambda_N \nabla c) = 0 \tag{5.25}$$

Thus, this part of the boundary is fixed. In the case of conditions (5.24), the nutrient concentration c is prescribed on $\Gamma_1(t)$. This means that nutrients can flow into and out of the domain. On $\Gamma_2(t)$, we impose boundary conditions

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\tau}, \quad p = 0, \quad \boldsymbol{n} \cdot (ck\nabla p + \lambda_N \nabla c) + \kappa(c - c_0) = 0. \tag{5.26}$$

Note that $\boldsymbol{\sigma}$ is the complete stress tensor given in (5.20). The parameter κ should be non-negative. Note that if $\kappa = 0$, we have a zero-flux boundary condition for c on $\Gamma_2(t)$. If in addition we use condition (5.25) on $\Gamma_1(t)$, nutrients can not flow into or out of the domain at any point.

(ii) This configuration is very similar to configuration (i). The same initial domain $\Omega(0) = (0,1)^2$ is used. On $\Gamma_1(t)$, we impose

$$\boldsymbol{v} = \boldsymbol{0}, \quad \boldsymbol{n} \cdot k\nabla p = 0, \quad \boldsymbol{n} \cdot (ck\nabla p + \lambda_N \nabla c) = 0, \tag{5.27}$$

and on $\Gamma_2(t)$ we impose

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} + \gamma \boldsymbol{u} = \boldsymbol{0}, \quad p = 0, \quad \boldsymbol{n} \cdot (ck\nabla p + \lambda_N \nabla c) = 0. \tag{5.28}$$

The parameter γ should be non-negative. If it is zero, then $\Gamma_2(t)$ is a free boundary and we are back in configuration (i) with $\boldsymbol{\tau} = \boldsymbol{0}$. The boundary condition on $\boldsymbol{\sigma}$ essentially models a mass-spring system: the force on the boundary is opposite and proportional to the displacement. This models the force of the surrounding tissue as it is pressing on the domain. The weak form of equation (5.17) must also be adapted, it now becomes

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega(t)} \rho \boldsymbol{v} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega + \int_{\Omega(t)} \boldsymbol{\sigma} : \nabla \boldsymbol{\varphi} \, \mathrm{d}\Omega + \int_{\Gamma_2(t)} \gamma \boldsymbol{u} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Gamma = \int_{\Omega(t)} \boldsymbol{g} \cdot \boldsymbol{\varphi} \, \mathrm{d}\Omega. \tag{5.29}$$

This leads to a semi-discrete system of the form

$$\frac{\mathrm{d}}{\mathrm{d}t} (\rho M_v \boldsymbol{v}) + S_{\mathrm{vis}} \boldsymbol{v} + S_{\mathrm{el}} \boldsymbol{\varepsilon} - D^\top \boldsymbol{p} + \widetilde{C} \boldsymbol{u} = \boldsymbol{g}, \tag{5.30}$$

where $\boldsymbol{u} = (\boldsymbol{u}^1, \boldsymbol{u}^2)^\top$ is the vector containing the domain displacement in the grid nodes, and $\widetilde{C}$ is the block matrix

$$\widetilde{C} = \begin{pmatrix} C & 0 \\ 0 & C \end{pmatrix},$$

where C is given element-wise by

$$C_{ij} = \gamma \int_{\Gamma_2(t)} \varphi_i \varphi_j \, \mathrm{d}\Gamma. \tag{5.31}$$

Note that the displacement does not appear explicitly in any of the other equations. We use the displacement velocity to update it:

$$\boldsymbol{u}^m = \boldsymbol{u}^{m-1} + \Delta t \boldsymbol{v}^m. \tag{5.32}$$

Thus, integrating (5.30) in time using implicit Euler yields

$$\begin{aligned} \left(\rho M_v^m + \Delta t S_{\mathrm{vis}}^m + \Delta t^2 \widetilde{C}^m\right) \boldsymbol{v}^m + \Delta t S_{\mathrm{el}}^m \boldsymbol{\varepsilon}^m - \Delta t (D^m)^\top \boldsymbol{p}^m \\ = \rho M_v^{m-1} \boldsymbol{v}^{m-1} - \Delta t \widetilde{C}^m \boldsymbol{u}^{m-1} + \Delta t \boldsymbol{g}^m \end{aligned} \tag{5.33}$$

(iii) Finally, we consider a quarter circle with radius R:

$$\Omega(0) = \{(x,y) : x, y > 0, \ x^2 + y^2 < R^2\},$$

with boundaries

$$\begin{aligned}\Gamma_{11}(0) &= \{(x,y) : 0 < x < R,\ y = 0\},\\ \Gamma_{12}(0) &= \{(x,y) : x = 0,\ 0 < y < R\},\\ \Gamma_{2}(0) &= \{(x,y) : x, y > 0,\ x^2 + y^2 = R^2\}.\end{aligned}$$

Figure 5.1 shows a sketch of this initial domain. The following boundary conditions are imposed:

$$\text{on } \Gamma_{11}(t): \quad v^2 = 0, \quad \boldsymbol{n} \cdot k\nabla p = 0, \quad \boldsymbol{n} \cdot (ck\nabla p + \lambda_N \nabla c) = 0. \tag{5.34}$$

$$\text{on } \Gamma_{12}(t): \quad v^1 = 0, \quad \boldsymbol{n} \cdot k\nabla p = 0, \quad \boldsymbol{n} \cdot (ck\nabla p + \lambda_N \nabla c) = 0. \tag{5.35}$$

$$\text{on } \Gamma_{2}(t): \quad \boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{0}, \quad p = 0, \quad c = c_b. \tag{5.36}$$

Thus, on $\Gamma_{11}(t)$, the domain can only move in the x-direction. Similarly, on $\Gamma_{12}(t)$, the domain can only move in the y-direction. The circle arc $\Gamma_2(t)$ is a free boundary. In this configuration, we are essentially modelling a full circle in which there are symmetries across its vertical and horizontal radius.

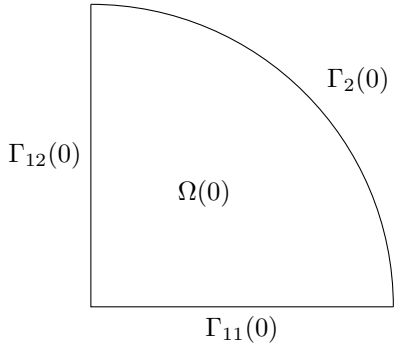


Figure 5.1: Sketch of initial domain with corresponding boundaries.

5.2 Analysis of Zero-Dimensional Nutrient Decay & Tissue Growth

In the following sections, equations (5.21) and (5.22) are analyzed in a zero-dimensional setting. We consider the oxygen concentration c, and later also the tumor volume fraction Φ, at a single point and assume there is no transport. Hence the evolution of these quantities only depends on their respective source functions and their coupling.

5.2.1 Analysis of Nutrient Absorption

We first investigate the nutrient absorption behaviour as modeled by the Michaelis-Menten equation. To this end, we analyze the following differential equation:

$$\frac{\mathrm{d}c}{\mathrm{d}t} = -\Phi N_{\mathrm{A}}(c) = -\frac{\Phi A_{\max} c}{c_A + c}, \tag{5.37}$$

where the concentration c is only a function of time: $c = c(t)$. This effectively mimics equation (5.21) in the case where $\boldsymbol{v} = \boldsymbol{0}$, ∇p is constant and $\lambda_N = 0$; there is no domain deformation, no Darcy flux and no nutrient diffusion respectively. In other words, there is no transport of oxygen and the evolution of the concentration is only determined by absorption. In this section, we assume Φ is constant in time and treat $\Phi A_{\max}$ as being one parameter. Later it will make sense to separate them, as Φ will no longer be constant in time. Equation (5.37) is separable:

$$\int \frac{c_A}{c} + 1 \,\mathrm{d}c = -\int \Phi A_{\max} \,\mathrm{d}t. \tag{5.38}$$

Which is solved by

$$c_A \ln|c| + c = -\Phi A_{\max} t + K, \tag{5.39}$$

where K is an integration constant. From the original differential equation it is clear that $\mathrm{d}c/\mathrm{d}t < 0$ for $c > 0$ and $\mathrm{d}c/\mathrm{d}t = 0$ if and only if $c = 0$. Therefore if $c(0) > 0$ we have $c(t) > 0$ for all $t \geqslant 0$. With this in mind we can drop the absolute value and eliminate K in favour of the (positive) initial condition $c(0) = c_0$:

$$c_A \ln\left(\frac{c(t)}{c_0}\right) + c(t) - c_0 = -\Phi A_{\max} t. \tag{5.40}$$

We can not solve the above equation for $c(t)$ in terms of elementary functions. However, we can either analyze the curve in t-c space defined by the implicit relation, or we can solve it using the Lambert W function [4]. This function is defined as the solution $W(z)$ to

$$W(z)e^{W(z)} = z. \tag{5.41}$$

The solution is uniquely determined for real $z > 0$. To solve equation (5.40) in terms of the Lambert W function, we will need the following result.

Lemma 9. *Let the function $f(z)$ be implicitly defined by*

$$f(z)^n e^{f(z)} = z, \tag{5.42}$$

for $z > 0$ and $n > 0$. Then

$$f(z) = nW\left(\frac{z^{1/n}}{n}\right). \tag{5.43}$$

Proof. We can prove the statement by simply subsituting (5.43) into (5.42):

$$\begin{aligned} f(z)^n e^{f(z)} &= \left(nW\left(\frac{z^{1/n}}{n}\right)\right)^n \exp\left[nW\left(\frac{z^{1/n}}{n}\right)\right] \\ &= \left(nW\left(\frac{z^{1/n}}{n}\right)\exp\left[W\left(\frac{z^{1/n}}{n}\right)\right]\right)^n \\ &= \left(n\frac{z^{1/n}}{n}\right)^n \\ &= z \end{aligned}$$

Here we have used definition (5.41). □

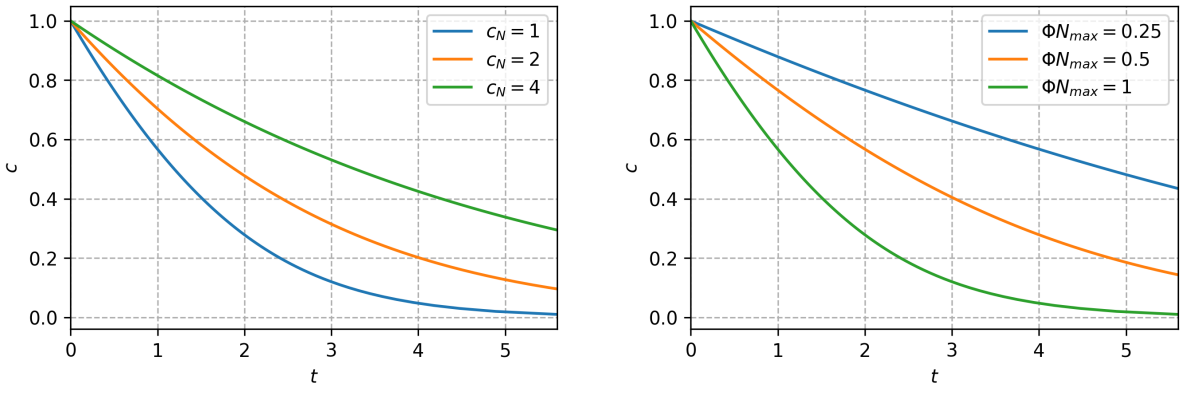
To apply the result of Lemma 9, we first write (5.40) in the form

$$c(t)^{c_A} e^{c(t)} = c_0^{c_A} \exp\left[c_0 - \Phi A_{\max} t\right], \tag{5.44}$$

by taking the exponential on both sides. It then follows that

$$c(t) = c_A W\left(\frac{c_0}{c_A}\exp\left[\frac{c_0 - \Phi A_{\max} t}{c_A}\right]\right) \tag{5.45}$$

Figure 5.2a and 5.2b show the effect of c_A and $\Phi A_{\max}$ on the evolution of the concentration. The other parameters are kept at 1 for the sake of demonstration.



(a) Concentration for various values of c_A.

(b) Concentration for various values of $\Phi A_{\max}$.

5.2.2 Coupling Nutrient Absorption and Volume Fraction Growth

To extend the model of the previous section, the solid volume fraction Φ is added to the system. We use equation (5.3) to model its growth or decay. Since c and Φ are assumed to only be functions of time, the stress dependent factor $F(\overline{\sigma})$ is assumed to be constant. Note that because there is no convective transport, Φ and η are equal. Hence, Φ is a measure of tissue growth. We get the following coupled system of differential equations:

$$\frac{\mathrm{d}c}{\mathrm{d}t} = -\Phi N_{\mathrm{A}}(c) \tag{5.46}$$

$$\frac{\mathrm{d}\Phi}{\mathrm{d}t} = F\Phi(1-\Phi)N_{\mathrm{G}}(c) - r_{\mathrm{d}}\Phi \tag{5.47}$$

The system is numerically integrated using the explicit Runge-Kutta 4 method. We use the initial conditions $c(0) = 1$ and $\Phi(0) = 0.01$ to simulate a situation in which there is initially almost no cell tissue present, such that the available nutrients will cause tissue growth. We set the cell creation- and nutrient absorption parameters to 1:

$$G_{\max} = c_G = A_{\max} = c_A = 1, \tag{5.48}$$

and we also set $F = 1$. The death rate r_{d} is varied to investigate the resulting behaviour. The system is integrated from $t = 0$ to $t = 100$ using a timestep of size $\Delta t = 0.1$. Figure 5.2 shows the solutions corresponding to $r_{\mathrm{d}} = 0$, $r_{\mathrm{d}} = 0.01$ and $r_{\mathrm{d}} = 0.1$.

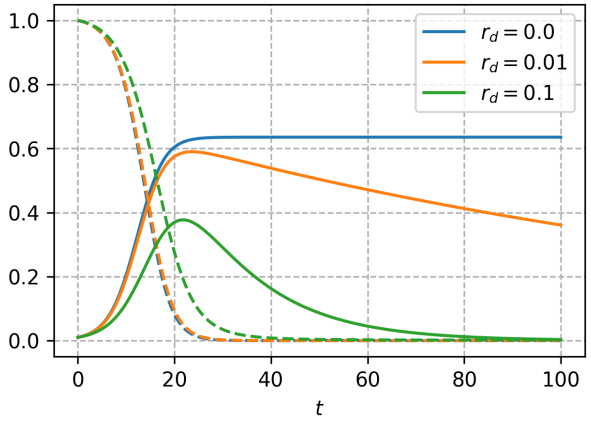


Figure 5.2: Plot of concentration (dashed line) and volume fraction (solid line) as a function of time for various values of r_{d}.

Initially Φ starts growing due to the high amount of nutrients. After the nutrient concentration gradually declines, growth slows down and the death rate takes over. In the case of $r_{\mathrm{d}} = 0$, Φ converges to a positive asymptotic value, whereas for $r_{\mathrm{d}} > 0$ it converges to 0.

5.3 Finite Element Approach

In this section,the weak form and corresponding Galerkin equations of equations (5.21), (5.22) and (5.23) are derived.

5.3.1 Nutrient Transport Equation

We start with the nutrient transport equation (5.21). Note that it is a convection-diffusion equation with a non-linear sink term. We can distinguish three types of terms in equation (5.21); a term accounting for the domain deformation, a concentration flux, and a concentration sink:

$$\underbrace{\frac{Dc}{Dt} + (\nabla \cdot \boldsymbol{v})c}_{\text{deformation}} \underbrace{- \nabla \cdot (kc\nabla p + \lambda_N \nabla c)}_{\text{flux}} = \underbrace{-\Phi N_{\mathrm{A}}(c)}_{\text{sink}}, \tag{5.49}$$

where the nutrient absorption rate $N_{\mathrm{A}}(c)$ is given in (5.15). For the derivation of the weak form we use a zero-flux boundary condition on the entire boundary:

$$\boldsymbol{n} \cdot (ck\nabla p + \lambda_N \nabla c) = 0. \tag{5.50}$$

This boundary condition is also imposed in testing configuration (ii) in Section 5.1.2. Note that if a Robin boundary condition is imposed, such as in configuration (i), then some additional terms will appear in the weak form. Let ξ be a Lagrangian basis function. Multiply the above equation by ξ and integrate over $\Omega(t)$ to obtain:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)} c\xi \,\mathrm{d}\Omega - \int_{\Omega(t)} \nabla\cdot(ck\nabla p + \lambda_N \nabla c)\,\xi \,\mathrm{d}\Omega = -\int_{\Omega(t)} \Phi N_{\mathrm{A}}(c)\xi \,\mathrm{d}\Omega. \tag{5.51}$$

Here we have applied Theorem 3 to obtain the first term. Use partial integration and Gauss' divergence theorem to find

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)} c\xi \,\mathrm{d}\Omega + \int_{\Omega(t)} ck\nabla p\cdot\nabla\xi + \lambda_N \nabla c\cdot\nabla\xi \,\mathrm{d}\Omega - \int_{\Gamma(t)} \boldsymbol{n}\cdot(ck\nabla p + \lambda_N\nabla c)\,\xi \,\mathrm{d}\Gamma = -\int_{\Omega(t)} \Phi N_{\mathrm{A}}(c)\xi \,\mathrm{d}\Omega. \tag{5.52}$$

Applying the boundary conditions yields

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)} c\xi \,\mathrm{d}\Omega + \int_{\Omega(t)} ck\nabla p\cdot\nabla\xi + \lambda_N \nabla c\cdot\nabla\xi \,\mathrm{d}\Omega = -\int_{\Omega(t)} \Phi N_{\mathrm{A}}(c)\xi \,\mathrm{d}\Omega. \tag{5.53}$$

This weak form introduces two difficulties: we obtain a non-linear term involving both c and p, and on the right-hand side we get a non-linear term involving Φ and a (non-linear) function of c. Using linear triangular elements, we can write

$$c(\boldsymbol{x},t) = \sum_{j=1}^{n} c_j(t)\varphi_j(\boldsymbol{x};t), \quad \Phi(\boldsymbol{x},t) = \sum_{k=1}^{n} \Phi_k(t)\varphi_k(\boldsymbol{x};t), \qquad p(\boldsymbol{x},t) = \sum_{\ell=1}^{n} p_\ell(t)\varphi_\ell(\boldsymbol{x};t), \tag{5.54}$$

where the φ_j are linear basis functions with respect to a mesh $\mathcal{T}_h$. Set $\xi = \varphi_i$ for some $i \in \{1,\dots,n\}$, then we get

$$\int_{\Omega(t)} ck\nabla p\cdot\nabla\xi + \lambda_N\nabla c\cdot\nabla\xi \,\mathrm{d}\Omega = \sum_{j=1}^{n} c_j \int_{\Omega(t)} k\varphi_j \sum_{\ell=1}^{n} p_\ell \nabla\varphi_\ell\cdot\nabla\varphi_i + \lambda_N\nabla\varphi_j\cdot\nabla\varphi_i \,\mathrm{d}\Omega. \tag{5.55}$$

Regarding the right-hand side, we approximate it in the following way:

$$\begin{aligned}
\int_{\Omega(t)} \Phi N_{\mathrm{A}}(c)\varphi_i \,\mathrm{d}\Omega &= \int_{\Omega(t)} \frac{A_{\max} c\Phi\varphi_i}{c_A + c} \,\mathrm{d}\Omega \\
&= \sum_{j=1}^{n} c_j \int_{\Omega(t)} \frac{A_{\max}\Phi\varphi_i\varphi_j}{c_A + c} \,\mathrm{d}\Omega \\
&\approx \sum_{j=1}^{n} c_j \sum_{T\in\mathcal{T}_h} \widetilde{A}_T(\boldsymbol{c}) \int_T \Phi\varphi_i\varphi_j \,\mathrm{d}\Omega, \\
&= \sum_{j=1}^{n} c_j \sum_{T\in\mathcal{T}_h} \widetilde{A}_T(\boldsymbol{c}) \sum_{\ell\in\{1,2,3\}} \Phi_{\ell,T} \int_T \varphi_{\ell,T}\varphi_i\varphi_j \,\mathrm{d}\Omega.
\end{aligned} \tag{5.56}$$

Here $\widetilde{A}_T(\boldsymbol{c})$ is an approximation of $A_{\max}/(c_A + c)$ on the element T:

$$\widetilde{A}_T(\boldsymbol{c}) = \frac{1}{3}\left(\frac{A_{\max}}{c_A + c_{1,T}} + \frac{A_{\max}}{c_A + c_{2,T}} + \frac{A_{\max}}{c_A + c_{3,T}}\right), \tag{5.57}$$

Here $\Phi_{\ell,T}$ and $c_{\ell,T}$ for $\ell \in \{1,2,3\}$ denote the values of Φ and c at the vertices of T. In conclusion, we get the following semi-discretized system of Galerkin equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}(M\boldsymbol{c}) + \big[kA(\boldsymbol{p}) + \lambda_N L + N_A(\boldsymbol{\Phi},\boldsymbol{c})\big]\boldsymbol{c} = \boldsymbol{0}, \tag{5.58}$$

where the coefficient matrices are given element-wise by:

$$M_{ij} = \int_{\Omega(t)} \varphi_i \varphi_j \, \mathrm{d}\Omega, \tag{5.59}$$

$$L_{ij} = \int_{\Omega(t)} \nabla\varphi_i \cdot \nabla\varphi_j \, \mathrm{d}\Omega, \tag{5.60}$$

$$A(\boldsymbol{p})_{ij} = \sum_{\ell=1}^{n} p_\ell \int_{\Omega(t)} \varphi_j \nabla\varphi_\ell \cdot \nabla\varphi_i \, \mathrm{d}\Omega. \tag{5.61}$$

$$N_A(\boldsymbol{\Phi}, \boldsymbol{c})_{ij} = \sum_{T\in\mathcal{T}_h} \widetilde{A}_T(\boldsymbol{c}) \sum_{\ell\in\{1,2,3\}} \Phi_{\ell,T} \int_T \varphi_{\ell,T}\varphi_i\varphi_j \, \mathrm{d}\Omega. \tag{5.62}$$

for $i, j \in \{1, \ldots, n\}$. The time-discretization is done in the usual way using implicit Euler:

$$M^m \boldsymbol{c}^m + \Delta t \left[k A^m(\boldsymbol{p}^m) + \lambda_N L^m + N_A^m(\boldsymbol{\Phi}^m, \boldsymbol{c}^m) \right] \boldsymbol{c}^m = M^{m-1}\boldsymbol{c}^{m-1}. \tag{5.63}$$

5.3.2 Volume Fraction Evolution

In this section, we discretize equation (5.22). For readability purposes, we repeat the equation:

$$\frac{D\Phi}{Dt} + (\nabla \cdot \boldsymbol{v})\Phi = F(\overline{\sigma})\Phi(1-\Phi)N_{\mathrm{G}}(c) - r_{\mathrm{d}}\Phi. \tag{5.64}$$

By multiplying the equation with φ_i for $i = 1, \ldots, n$ and integrating over $\Omega(t)$, we obtain a system of the form

$$\frac{\mathrm{d}}{\mathrm{d}t}(M\boldsymbol{\Phi}) = \boldsymbol{\mathcal{S}} - r_{\mathrm{d}} M\boldsymbol{\Phi}. \tag{5.65}$$

The first term is again the result of applying Theorem 3. Moveover, the vector $\boldsymbol{\mathcal{S}}$ is given element-wise by:

$$\boldsymbol{\mathcal{S}}_i = \sum_{T\in\mathcal{T}_h} F(\overline{\sigma}_T)\widetilde{G}_T(\boldsymbol{c}) \sum_{j=1}^{n} c_j \left[\sum_{k\in\{1,2,3\}} \Phi_{k,T} \int_T \varphi_i\varphi_j\varphi_{k,T} \, \mathrm{d}\Omega \right.$$
$$\left. - \sum_{k\in\{1,2,3\}} \sum_{\ell\in\{1,2,3\}} \Phi_{k,T}\Phi_{\ell,T} \int_T \varphi_i\varphi_j\varphi_{k,T}\varphi_{\ell,T} \, \mathrm{d}\Omega \right]. \tag{5.66}$$

The element matrices corresponding to the integrals in the above expression, as well as in (5.56), are computed in Section A.4. Similar to (5.57), we use

$$\widetilde{G}_T(\boldsymbol{c}) = \frac{1}{3}\left(\frac{G_{\max}}{c_G + c_{1,T}} + \frac{G_{\max}}{c_G + c_{2,T}} + \frac{G_{\max}}{c_G + c_{3,T}} \right) \tag{5.67}$$

as an approximation of $\widetilde{G}(c) = G_{\max}/(c + c_G)$ on T. The factor $\overline{\sigma}_T$ is the average over a triangle T of the total stress (5.13). Thus we have

$$\overline{\sigma}_T = \sum_{\ell\in\{1,2,3\}} \left[\frac{1}{3}\left(\frac{\mu_\varepsilon}{2} + \lambda_\varepsilon\right)\left(\varepsilon^{11}_{\ell,T} + \varepsilon^{22}_{\ell,T}\right) + \left(\frac{\mu_v}{2} + \lambda_v\right)\left(v^1_{\ell,T}\alpha_{\ell,T} + v^2_{\ell,T}\beta_{\ell,T}\right) \right.$$
$$\left. - \frac{1}{3}p_{\ell,T} - \frac{1}{3}\left(\frac{\mu_\varepsilon}{3} + \lambda_\varepsilon\right)\eta_{\ell,T} \right], \tag{5.68}$$

where $\alpha_{\ell,T}$ and $\beta_{\ell,T}$ are respectively the x- and y-derivative of the linear basis function $\varphi_{\ell,T}$, and a subscript (ℓ, T) denotes one of the three vertices of T. Discretizing the semi-discrete system (5.65) in time using implicit Euler yields:

$$(1 + \Delta t r_{\mathrm{d}})M^{m+1}\boldsymbol{\Phi}^{m+1} - \Delta t F \boldsymbol{s}^{m+1}(\boldsymbol{\Phi}^{m+1}, \boldsymbol{c}^{m+1}) = M^m\boldsymbol{\Phi}^m \tag{5.69}$$

5.3.3 Updating New Tissue Volume

In this section, we will discuss the treatment of the new tissue volume η. Note that equations (5.22) and (5.23) are very similar. The evolution of Φ takes into account the domain deformation, whereas the evolution of η does not. This corresponds to their physical interpretations; Φ represents the fraction of space occupied by tumor cells, whereas η represents the volume of new tumor cells *per unit of volume*. Thus, where Φ is affected by the flow of the domain, η is not. We derive the weak form of equation (5.23). Since we are using a Lagrangian framework - the value of the unknowns is tracked at the location of a moving grid point - the partial time-derivative $\partial\eta/\partial t$ cannot be easily discretized, because we must compensate for the displacement velocity $\boldsymbol{v}$. To this end, let ξ be a scalar-valued linear Lagrangian basis function, then:

$$\begin{aligned}\int_{\Omega(t)} \frac{\partial \eta}{\partial t}\xi \,\mathrm{d}\Omega &= \int_{\Omega(t)} \left(\frac{\partial \eta}{\partial t} + \nabla\cdot(\boldsymbol{v}\eta)\right)\xi \,\mathrm{d}\Omega - \int_{\Omega(t)} \nabla\cdot(\boldsymbol{v}\eta)\xi \,\mathrm{d}\Omega \\ &= \frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega(t)} \eta\xi \,\mathrm{d}\Omega - \int_{\Omega(t)} \left((\nabla\cdot\boldsymbol{v})\eta + \boldsymbol{v}\cdot\nabla\eta\right)\xi \,\mathrm{d}\Omega. \end{aligned} \tag{5.70}$$

Note that we added and subtracted the term $\nabla\cdot(\boldsymbol{v}\eta)\xi$. This then yields an integral to which Theorem 3 can be applied. We now write

$$\eta(\boldsymbol{x},t) = \sum_{j=1}^{n} \eta_j(t)\varphi_j(\boldsymbol{x};t),$$

where φ_j is a linear basis function centered in grid node j. Moreover, let $\boldsymbol{\eta}$ be the vector containing the η_j. Set $\xi = \varphi_i$ for $i = 1,\ldots,n$ to obtain the follwing semi-discrete system for (5.23):

$$\frac{\mathrm{d}}{\mathrm{d}t}(M\boldsymbol{\eta}) - Q(\boldsymbol{v})\boldsymbol{\eta} = \boldsymbol{\mathcal{S}} - r_{\mathrm{d}}M\boldsymbol{\Phi}, \tag{5.71}$$

where the vector $\boldsymbol{\mathcal{S}}$ is given element-wise by (5.66), and

$$Q(\boldsymbol{v})_{ij} = \int_{\Omega(t)} (\nabla\cdot\boldsymbol{v})\varphi_i\varphi_j + (\boldsymbol{v}\cdot\nabla\varphi_j)\varphi_i \,\mathrm{d}\Omega. \tag{5.72}$$

The per-element contributions of the matrix $Q(\boldsymbol{v})$ are given by:

$$(Q(\boldsymbol{v})_T)_{ij} = \sum_{\ell\in\{1,2,3\}} \left[\left(v^1_{\ell,T}\alpha_{\ell,T} + v^2_{\ell,T}\beta_{\ell,T}\right)\int_T \varphi_i\varphi_j \,\mathrm{d}\Omega + \left(v^1_{\ell,T}\alpha_j + v^2_{\ell,T}\beta_j\right)\int_T \varphi_{\ell,T}\varphi_i \,\mathrm{d}\Omega\right]. \tag{5.73}$$

Applying implicit Euler to (5.71) yields

$$\left(M^m - \Delta t Q^m(\boldsymbol{v}^m)\right)\boldsymbol{\eta}^m - \Delta t\boldsymbol{\mathcal{S}}^m + \Delta t r_{\mathrm{d}}M^m\boldsymbol{\Phi}^m = M^{m-1}\boldsymbol{\eta}^{m-1} \tag{5.74}$$

Recall that $\boldsymbol{\Phi}$ satisfies the semi-discrete system (5.65). By subtracting this equation from (5.71), we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(M(\boldsymbol{\eta} - \boldsymbol{\Phi})\right) = Q(\boldsymbol{v})\boldsymbol{\eta}. \tag{5.75}$$

Thus, if $\boldsymbol{v}$ remains small in both magnitude and gradient, the term on the right-hand side will also be small. In other words, if the domain deforms slowly, we expect $\eta - \Phi$ to remain approximately constant in time. Moreover, if their initial values are equal, i.e. $\boldsymbol{\eta}^0 = \boldsymbol{\Phi}^0$, then η and Φ will be roughly equal on a slowly deforming domain. This is illustrated in Figure 5.3. Here we have applied a body force $\boldsymbol{g} = (1,0)^\top$ to speed up the domain deformation. Both η and Φ have been set to the constant initial value 0.5. It is clear that the difference between η and Φ is very small, with the maximum error being less than 1% relative to both η and Φ.

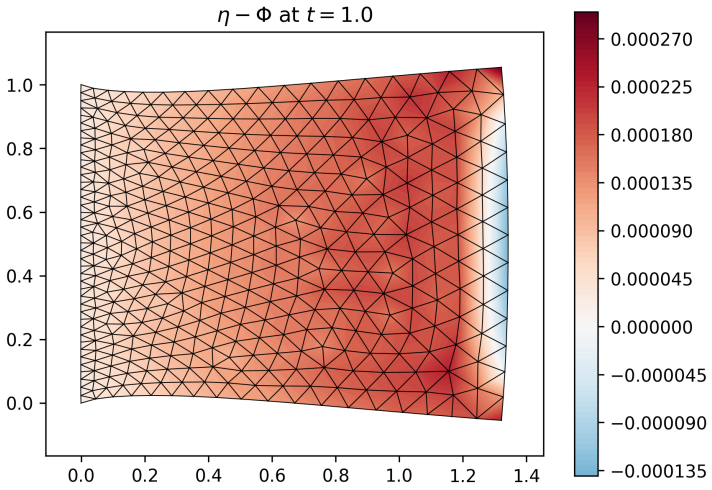


Figure 5.3: Contour plot of $\eta - \Phi$ after 10 timesteps, superimposed on the deformed grid.

5.3.4 Summary of Finite Element Model

We present the complete finite element tumor growth model. First, consider the semi-discrete mechanical model corresponding to the partial differential equations (5.17) - (5.19):

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\rho M_v \boldsymbol{v}\right) + S_{\mathrm{vis}}\boldsymbol{v} + S_{\mathrm{el}}\boldsymbol{\varepsilon} - D^\top \boldsymbol{p} - \left(\frac{\mu_\varepsilon}{3} + \lambda_\varepsilon\right) D^\top \boldsymbol{\eta} = \boldsymbol{g}, \tag{5.76}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(M_\varepsilon \boldsymbol{\varepsilon}\right) - B\boldsymbol{v} + \alpha M_\varepsilon \boldsymbol{\varepsilon} + \boldsymbol{\mathcal{N}}(\boldsymbol{v}, \boldsymbol{\varepsilon}) = \boldsymbol{0}, \tag{5.77}$$

$$D\boldsymbol{v} + kL\boldsymbol{p} = \boldsymbol{f}. \tag{5.78}$$

Here $\boldsymbol{v}$ and $\boldsymbol{\varepsilon}$ are the vectors $\boldsymbol{v} = \left(\boldsymbol{v}^1, \boldsymbol{v}^2\right)^\top$ and $\boldsymbol{\varepsilon} = \left(\boldsymbol{\varepsilon}^{11}, \boldsymbol{\varepsilon}^{12}, \boldsymbol{\varepsilon}^{22}\right)^\top$. The mass matrices M_v and M_ε denote block matrices with respectively 2 and 3 standard mass matrices M on their block-diagonals, and zeroes everywhere else. The coupled semi-discrete biochemical model, corresponding to equations (5.21) - (5.23), is given by:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(M\boldsymbol{c}\right) + \left[kA(\boldsymbol{p}) + \lambda_N L + N_A(\boldsymbol{\Phi}, \boldsymbol{c})\right]\boldsymbol{c} = \boldsymbol{0} \tag{5.79}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(M\boldsymbol{\Phi}\right) + r_{\mathrm{d}} M\boldsymbol{\Phi} = \boldsymbol{\mathcal{S}} \tag{5.80}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(M\boldsymbol{\eta}\right) - Q(\boldsymbol{v})\boldsymbol{\eta} + r_{\mathrm{d}} M\boldsymbol{\Phi} = \boldsymbol{\mathcal{S}} \tag{5.81}$$

The vector $\boldsymbol{\mathcal{S}}$ depends on all unknowns; $\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{S}}(\boldsymbol{v}, \boldsymbol{\varepsilon}, \boldsymbol{p}, \boldsymbol{c}, \boldsymbol{\Phi}, \boldsymbol{\eta})$. We have omitted this dependence in the above equations to increase readability. As was previously discussed, the two systems are solved simultaneously: we use a monolithic approach. The time-discretization is again done using implicit Euler. We will not give the fully discrete equations since they follow immediately from (5.76) - (5.81), and have already been shown in the previous sections. Note that due to the moving domain and the many non-linearities in the equations, an iterative method must be used to perform one timestep. We use a Picard-type iterative scheme, for more details see Section 2.4.3.

It should be noted that depending on the boundary conditions, additional terms must be added to one or more of the above equations. For example, if configuration (i) from Section 5.1.2 is used with $\kappa > 0$, equation (5.79) will gain additional terms involving boundary integrals over $\Gamma_2(t)$. Similarly, in configuration (ii), equation (5.76) will gain an additional term involving the boundary integral of the displacement, as is shown in (5.29).

5.4 Numerical Experiments

The tumor growth model is tested using the configurations from Section 5.1.2. The numerical experiments in this section should be considered as demonstrations of the finite element model. The simulations are not yet intended to model biologically correct tumor growth.

5.4.1 Configuration (i)

Domain Deformation Resulting from Tumor Cells

For our first experiment, we are interested in the effect of Φ on the domain deformation. The experiment is performed in the framework of configuration (i). We choose the following initial profile for Φ:

$$\Phi(x, y, 0) = 0.01 + 0.98 \cdot \mathbb{1}_{[0.5,1]}(x)(x - 0.5) \tag{5.82}$$

Here $\mathbb{1}_A$ denotes the indicator function of the set A. For $x < 0.5$, Φ has the constant value of 0.01. It is deliberately not set to zero, because then the source function S would vanish on this part of the domain. For $x \geqslant 0.5$, Φ increases linearly to a value of 0.5 at $x = 1$. Note that this means that there are already tumor cells present. The initial concentration profile is set to the constant value 1. We expect the gradients to be higher on the right side of the domain. Therefore, the following variable initial meshwidth is used:

$$h(x, y) = 0.03 + 0.07(1 - x) \tag{5.83}$$

Figure 5.4 shows the initial volume fraction and initial finite element mesh.

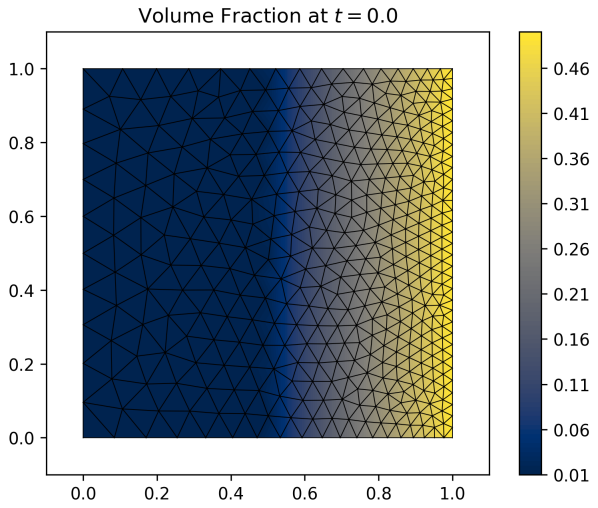


Figure 5.4: Contour plot of initial volume fraction profile (5.82) and initial grid using the variable meshwidth (5.83).

We run the morpho-visco-poroelastic tumor growth model using the simulation parameters in Tables 5.1, 5.2, 5.3.

Table 5.1: Mechanical parameters.

ρ	μ_ε	λ_ε	μ_v	λ_v	k	α
1	1	1	1	1	1	1

Table 5.2: Biological parameters.

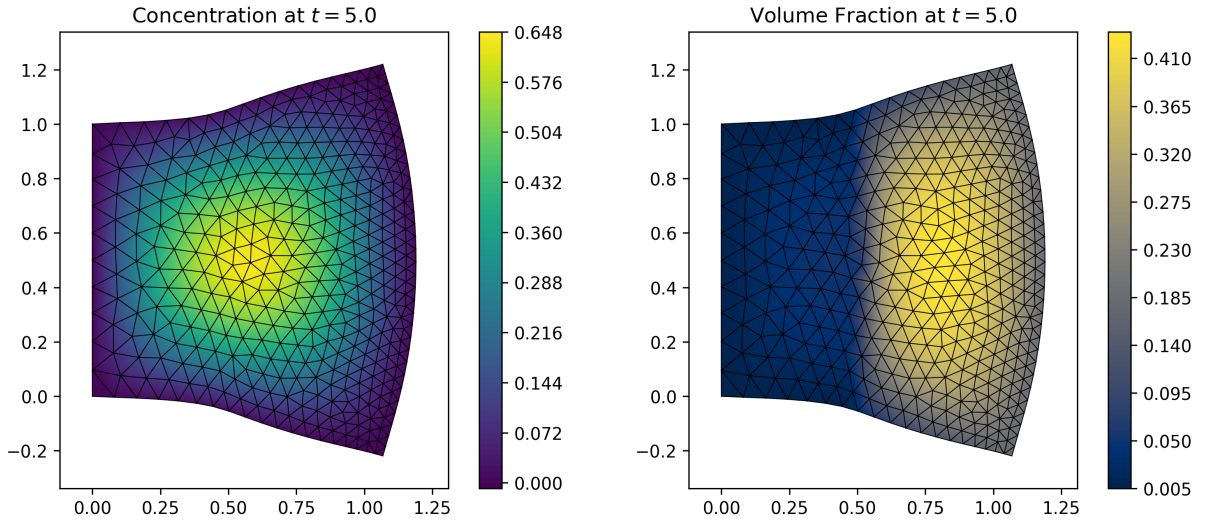
λ_N	$A_{\max}$	c_A	$G_{\max}$	c_G	r_{d}
0.01	1	1	1	1	0.1

Table 5.3: Source functions and boundary conditions.

$\boldsymbol{g}$	f	$\boldsymbol{\tau}$	c_0	c_b	κ
$\mathbf{0}$	0	$\mathbf{0}$	0	0	1

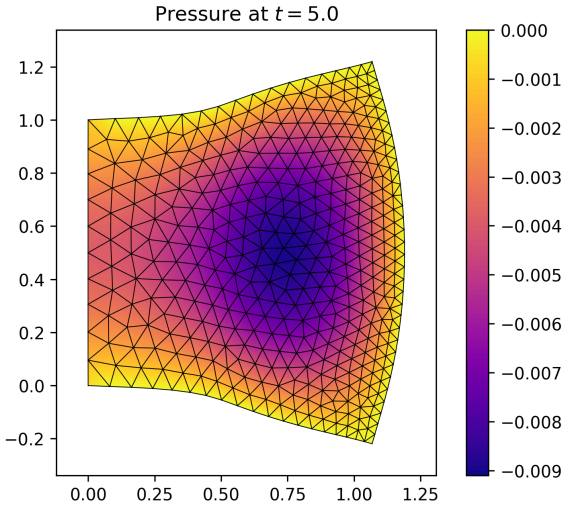
The model is run until $t = 5$ using a timstep of size $\Delta t = 0.1$. Figures 5.5a and 5.5b show the finite element solutions for c and Φ. Figure 5.5c shows the pressure solution. It is important to note that

there is no body force or shear stress applied to the medium; all deformations are a direct result of the prescence of tumor cells.



(a) Concentration.

(b) Volume fraction.



(c) Pressure.

Figure 5.5: Finite element solutions for c and Φ, superimposed on the deformed grid at $t = 5$. The corresponding simulation parameters are shown in Tables 5.1, 5.2 and 5.3.

Measuring Total Concentration & Volume Fraction

For our second experiment we are interested in the evolution of the total concentration c_{tot} and total volume fraction Φ_{tot} as functions of time. There are defined as:

$$c_{\text{tot}}(t) = \int_{\Omega(t)} c \, \mathrm{d}\Omega, \quad \Phi_{\text{tot}}(t) = \int_{\Omega(t)} \Phi \, \mathrm{d}\Omega. \tag{5.84}$$

The initial domain is again $\Omega(0) = (0, 1)^2$. We use a constant initial volume fraction/new tissue volume of $\Phi(x, y, 0) = \eta(x, y, 0) = 0.01$, and the initial concentration profile is given by

$$c(x, y, 0) = 16xy(1 - x)(1 - y). \tag{5.85}$$

We use the following simulation parameters:

Table 5.4: Mechanical parameters.

ρ	μ_ε	λ_ε	μ_v	λ_v	k	α
1	1	1	1	1	1	0.1

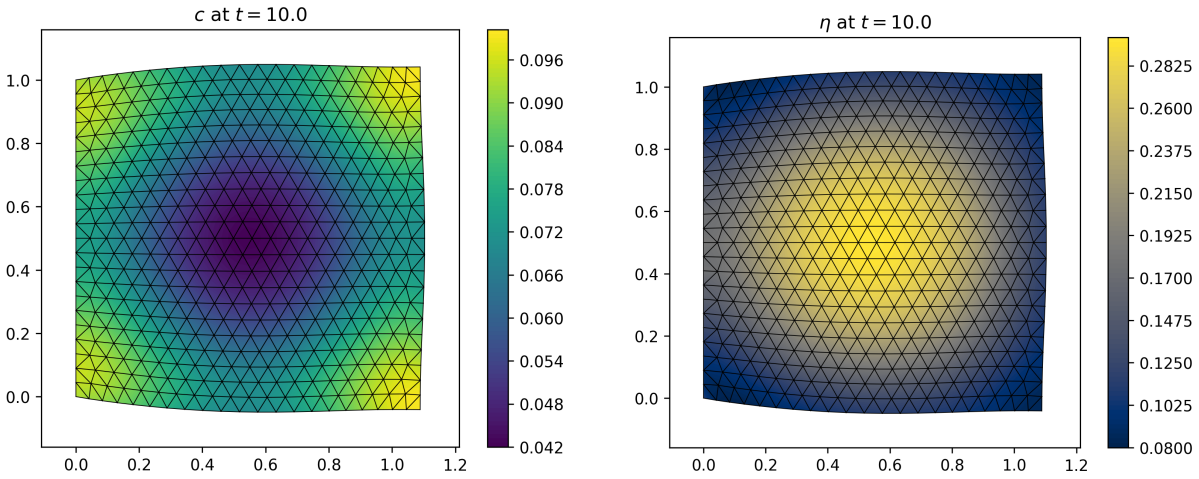
Table 5.5: Biological parameters.

λ_N	$A_{\max}$	c_A	$G_{\max}$	c_G	r_{d}
0.01	2	1	2	1	0.1

Table 5.6: Source functions and boundary conditions.

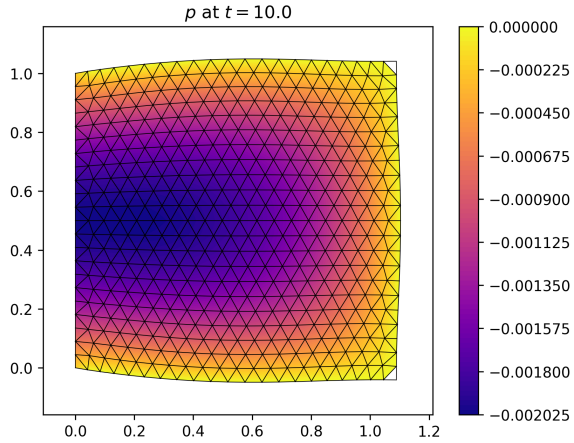
$\boldsymbol{g}$	f	$\boldsymbol{\tau}$	c_0	c_b	κ
$\mathbf{0}$	0	$\mathbf{0}$	0	0	0

A constant (initial) meshwidth of $h = 0.05\sqrt{2}$ is used. We run the model from $t = 0$ to $t = 50$ using a timestep of size $\Delta t = 0.25$. Figures 5.6a, 5.6b and 5.6c show the concentration, new tissue volume and pressure respectively at $t = 10$.



(a) Concentration.

(b) New tissue volume.



(c) Pressure.

Figure 5.6: Contour plots of finite element solutions for c, η and p, superimposed on the deformed grid at $t = 10$. The corresponding simulation parameters are shown in Tables 5.4, 5.5 and 5.6.

Figure 5.7 shows the evolution of the total concentration, total new tissue volume and domain area. Note that the evolution of c_{tot} and η_{tot} look qualitatively similar to that of c and Φ in the zero-dimensional case, see Figure 5.2. Moreover, because we use $\alpha = 0.1$, the domain has become permanently deformed. Thus, even when the tumor cells have all died, the domain does not bounce back to its original state.

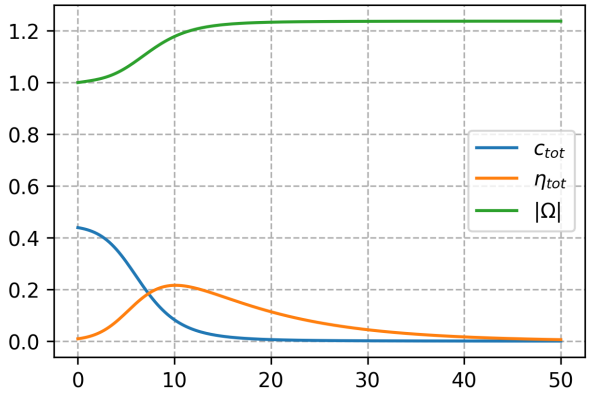


Figure 5.7: Total concentration, total new tissue volume and domain area as functions of time.

5.4.2 Configuration (ii)

Recall that compared to configuration (i), in configuration (ii) we replace the condition $\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{\tau}$ with $\boldsymbol{\sigma} \cdot \boldsymbol{n} + \gamma \boldsymbol{u} = 0$ on $\Gamma_2(t)$. To investigate the impact of the 'pushback' force, the same initial domain, initial conditions and simulation parameters are used as in the previous section. They can be found in Tables 5.4, 5.5 and 5.6. Figure 5.8 shows the effect of γ on the evolution of the domain area. Note that for $\gamma = 0$, there is no shear force acting on the boundary as a result of the displacement. For higher values of γ, the shear force becomes larger, prohibiting growth and causing the domain to converge to its original state after η has vanished.

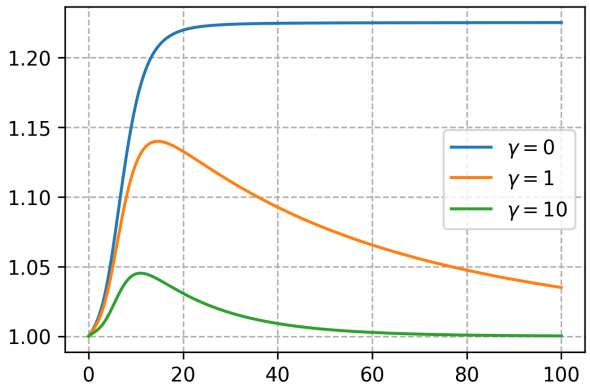


Figure 5.8: Domain area as a function of time for various values of γ.

5.4.3 Configuration (iii)

We test configuration (iii) using a quarter circle of radius $R = 1$. We use the following initial concentration profile:

$$c(0, x, y) = \frac{5}{\pi} \left(1 - (x^2 + y^2)^4\right) \left(1 + \sin(4 \arctan(y/x))\right). \tag{5.86}$$

The factor $5/\pi$ ensures that $c_{\text{tot}}(0) = 1$. Moreover, note that $\arctan(y/x)$ equals the angular coordinate φ in polar coordinates. Thus, we can also denote this concentration by:

$$c(0, r, \varphi) = \frac{5}{\pi} \left(1 - r^8\right) \left(1 + \sin(4\varphi)\right). \tag{5.87}$$

Figure 5.9 shows a contour plot of the above function. Note that for $\varphi < \pi/4$ there is a higher concentration than for $\varphi > \pi/4$. We set $c = 0$ on the free boundary $\Gamma_2(t)$. The term $1 - r^8$ ensures that the

initial concentration goes smoothly to its boundary value, whilst at the same time keeping the oscillating behaviour of the sine function in the interior of the domain.

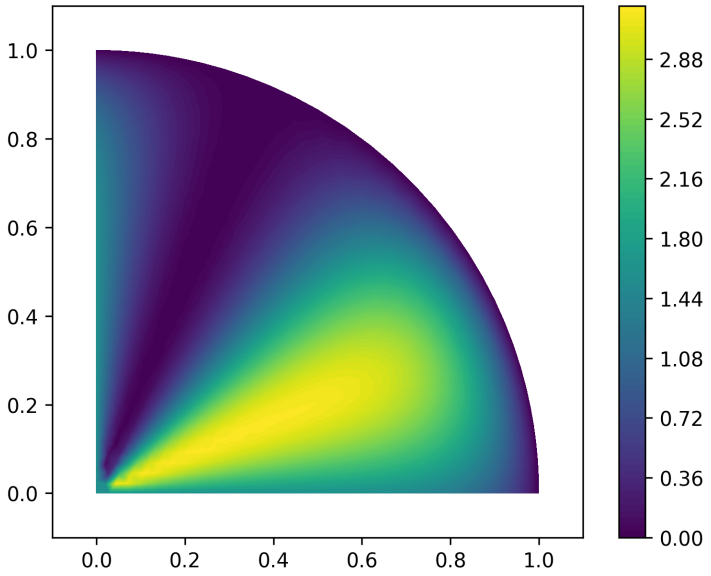


Figure 5.9: Contour plot of initial concentration (5.86)

Both Φ and η are set to the constant value of 0.01 on the initial domain. The simulation is performed until $t = 25$, using the parameters from Tables 5.7, 5.8 and 5.9, and a timestep of size $\Delta t = 0.25$. A maximum initial meshwidth of $h = 0.05\sqrt{2}$ is used. Figure 5.10 shows the domain area, total new tissue volume and total concentration as a function of time. Note that at $t = 25$ both η_{tot} and c_{tot} are nearly zero. However, the domain area seems to converge to a larger value than its initial area. This is due to the inclusion of morphoelasticity by setting $\alpha = 0.1$. Although nearly all tumor cells have died, they have resulted in a permanent deformation in the tissue.

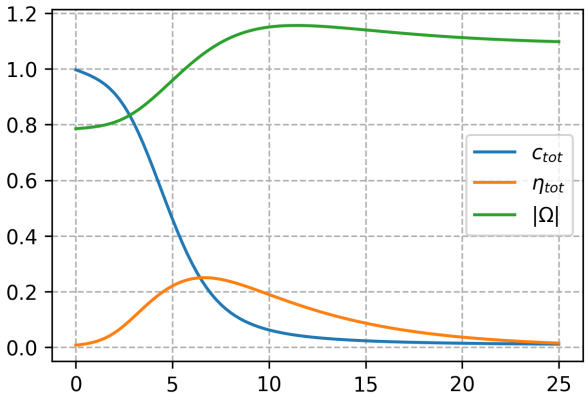


Figure 5.10: Domain area $|\Omega|$, total new tissue volume η_{tot} and total oxygen concentration c_{tot} as a function of time.

Figure 5.11 shows the strains at the halfway point of the simulation. Note that the domain is the most deformed near the x-axis. This is the result of tumor cells growing most rapidly at this position.

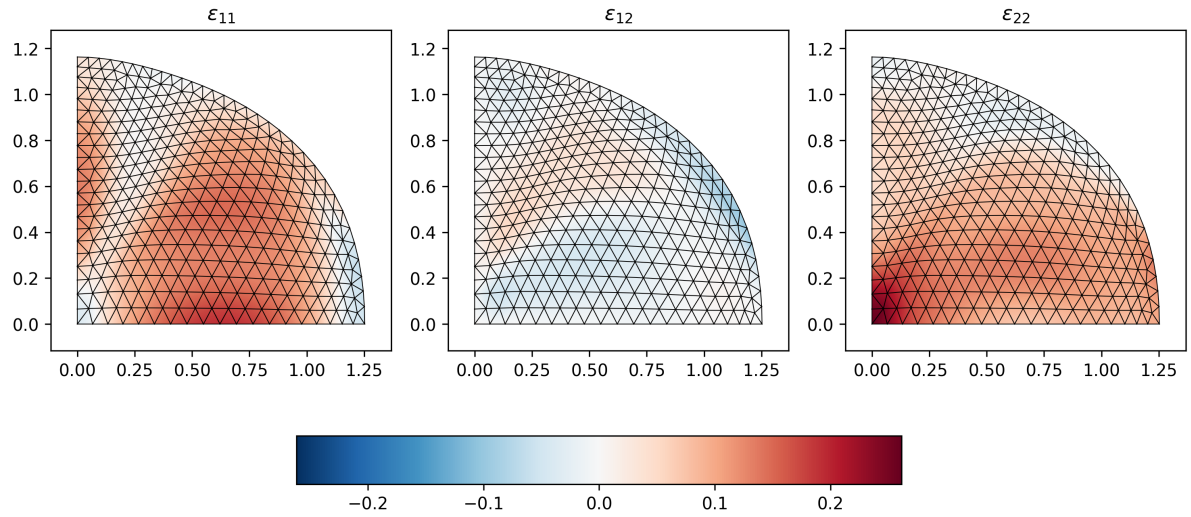


Figure 5.11: Components of the strain tensor at $t = 12.5$ superimposed on the deformed grid.

An interesting way to visualize the growth of the domain is to consider the distribution of its radius. We view the outer boundary $\Gamma_2(t)$ as the graph of an evolving function in polar coordinates:

$$\Gamma_2(t) = \left\{ (r(\varphi, t)\cos(\varphi), r(\varphi, t)\sin(\varphi)) \mid \varphi \in \left[0, \frac{\pi}{2}\right] \right\}. \tag{5.88}$$

Figure 5.12 shows the function $r(\varphi, t)$ for various values of t.

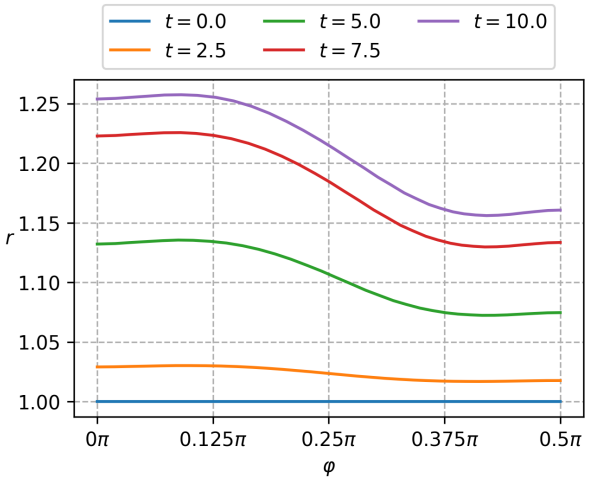
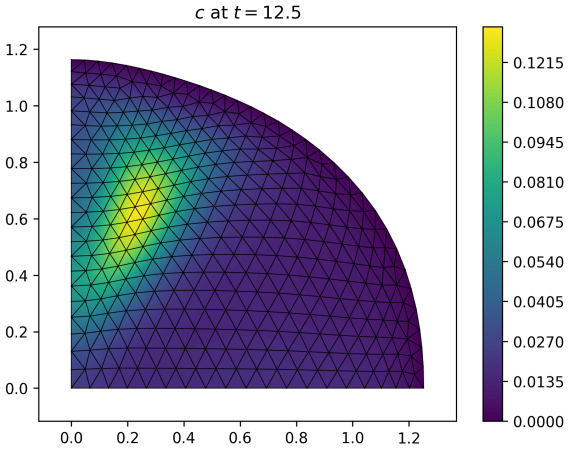
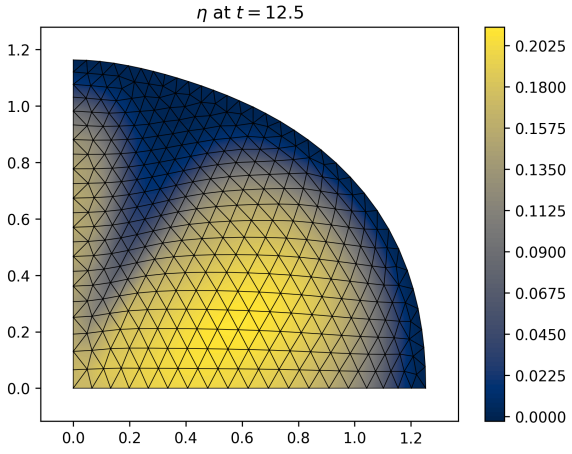


Figure 5.12: Distribution of the radius of the domain at various points in time.

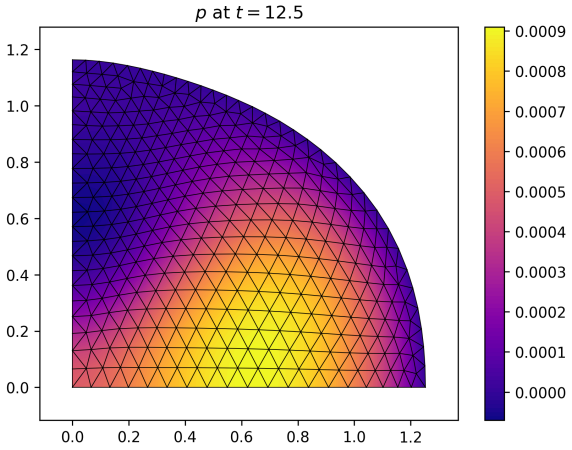
Figures 5.13a, 5.13b and 5.13c show contour plots of the concentration c, new tissue volume η and pressure p at the halfway point of the simulation. Note that the concentration peak is now located at the top left side of the domain. At $t = 0$, there was a concentration dip in this area. Thus, no tumor cells have developed here and hence oxygen is absorbed at a slower rate.



(a) Concentration.



(b) New tissue volume.



(c) Pressure.

Figure 5.13: Contour plots of finite element solutions for c, η and p, superimposed on the deformed grid at $t = 12.5$. The corresponding simulation parameters are found in Tables 5.7, 5.8 and 5.9.

The simulation parameters are presented in the following tables.

Table 5.7: Mechanical parameters.

ρ	μ_ε	λ_ε	μ_v	λ_v	k	α
0.25	0.5	0.5	1	1	1	0.1

Table 5.8: Biological parameters.

λ_N	$A_{\max}$	c_A	$G_{\max}$	c_G	r_{d}
0.001	2	1	2	1	0.2

Table 5.9: Source functions and boundary conditions.

$\boldsymbol{g}$	f	c_b
$\mathbf{0}$	0	0

6

Conclusion

In this work a finite element tumor growth model is developed, based on the combination of morpho-poroelasticity and a biochemcial model. The aim of Part I of this thesis is to gain insight in the behaviour of various elastic and poroelastic models. Chapter 2 started with a two dimensional finite element model for pure elasticity. This model is then expanded to viscoelasticity, in which the elastic material is assumed to possess some viscous properties. The viscoelastic model is solved using a Lagrangian approach, in other words: a moving mesh is used. To compute and simplify the weak forms in a Lagrangian setting, Theorem 3 is used extensively throughout this work. The proof of this theorem relies on Reynold's transport theorem, and a result regarding Lagrangian basis functions. An important characteristic of the solution to a viscoelastic model is its 'overshoot' property: the domain oscillates around a stationary solution.

Next, morphoelasticity is introduced. In a morphoelastic framework the deformation tensor is decomposed in an elastic and plastic part. The plastic part causes permanent deformations in the material. The decomposition leads to an evolution equation for the strain tensor. Hence, an important difference between a visco- and morphoelastic model is that the displacement vector is no longer explicitly obtained in the latter model. To obtain the displacement vector, which is necessary to update the mesh, postprocessing is applied in every timestep. Solutions to the morphoelastic model also exhibit the 'overshoot' property, but converge to a different stationary solution due to permanent deformation in the material.

Chapter 3 introduces the concept of poroelasticity. In this framework the material is assumed to be porous and completely saturated with a poro-fluid. The presence of this fluid means that the fluid pressure must be included in the force balance. A well-known problem occuring in poroelastic modeling is the emergence of non-physical oscillations in the fluid pressure. To explore why this happens we first consider a one-dimensional simplified poroelastic model called the Terzaghi problem. We find that oscillations occur for timesteps smaller than a certain lower bound that is dependent on the order of the elements. We show that for the Terzaghi problem, these oscillations can be completely fixed by introducing a diffusive stabilization matrix.

In the two-dimensional poroelastic model these oscillations also occur for small timesteps, and for small values of the parameter k, which is the ratio between the dynamic viscosity of the fluid and the permeability of the medium. Two methods of stabilization are compared: bubble functions and diffusive stabilization. Some oscillations are still present near the domain boundaries when using bubble functions. Diffusive stabilization seems more promising, but requires the user to specify a tuning parameter. Lastly, a visco-poroelastic model is developed. Interestingly, in this framework the oscillations occur only for small values of k, and not for small timesteps.

In Part II of this thesis we show some novel results. First, the visco-poroelastic model is analyzed in order to obtain an approximately optimal tuning parameter for the diffusive stabilization. This tuning parameter is optimal in the sense that it is the smallest parameter for which the evolution matrix for the pressure is again an M-matrix. The key step of the derivation is finding an approximate inverse of the discrete version of a reaction-Laplacian operator. This is achieved by considering a closely related boundary value problem with a delta function in the right-hand side. Error bounds are proved using

functional analytic methods. The optimal tuning parameter turns out to be very similar to those found in the literature on poroelasticity.

Another novel result is the combination of morpho-and poroelasticity. We develop a finite element model that combines the permanent deformations of morphoelasticity with poroelastic behaviour. Moreover, it is shown that that when k approaches infinity, the solution to the morpho-poroelastic system tends to the solution of the morphoelastic system. However a more rigorous proof of this fact is desired. Using the same method as for the visco-poroelastic system, an approximately optimal tuning parameter can also be obtained for diffusive stabilization of the morpho-poroelastic system.

In the final chapter, the morpho-poroelastic model is combined with a biochemical tumor growth model developed by Roose et al. in [14]. This introduces three new unknowns into the system: the oxygen concentration, the tumor cell volume fraction and the new tissue volume. The oxygen concentration satisfies a convection-diffusion equation, where the convective transport is driven by the fluid pressure. The other two unknowns satisfy two very similar evolution equations with a non-linear source function. The tumor cell volume fraction takes into account the domain deformation, whereas the new tissue volume does not. We see that for slowly deforming domains, these two unknowns are almost equal. The qualitative behaviour of the oxygen concentration and tumor volume fraction is analyzed in a zero-dimensional setting. Finally, the tumor growth model is tested in a number of different testing configurations.

6.1 Suggestions for Further Work

In order to improve the model or further develop the mathematical theory, we put forward the following suggestions for futher work:

- We suggest further research should be done to investigate the relation between the strain in a tissue and the degree of pain a patient experiences. We hypothesize that these two quantities are more or less proportional. This question introduces the difficulty of quantifying the strain and the amount of pain, the latter of which is particularly subjective.

- At a certain point in a tumor's life, it becomes so large that the surrounding tissue collapses. When this happens, tumor cells can enter the circulatory system and form colonies at other places in the body. This process is called metastasis. A possible way to formalize this 'breaking point' is to consider the work done by the tumor boundary on the surrounding tissue. Metastasis occurs whenever this energy reaches a threshold value. The finite element model can be easily adapted to compute the energy, but additional research is necessary to verify the validity of this approach.

- In Section 4.5.1 we present the argument that the solution to the morpho-poroelastic system approaches the solution to the morphoelastic system as k tends to infinity. In the proof of this claim some assumptions are made without being proved. Most notably, it is assumed that $\boldsymbol{v}$ stays bounded. While this seems like a reasonable assumption since $\boldsymbol{v}$ is an H^1-function, it should be proved in future work to make the argument more rigorous.

- A possible way to expand the model is to include additional nutrients. For example, it is known that next to oxygen, tumor growth is also dependent on the presence of glucose. By including glucose transport in the model the results will become more realistic. On the other hand, one could argue that oxygen and glucose are both transported by the same intercellular fluid, and that the growth rate depends on both nutrients in a similar fashion. Thus, we predict the model output will not be drastically different.

- An obvious improvement to the model would be to use higher order elements. On one hand, this will lead to a more accurate solution, but on the other hand the systems will grow in size. We have shown in Chapter 3 that Taylor-Hood elements lead to a slightly more stable solution, even without stabilization. Additionally, it could be beneficial to investigate the order of the error made by the model. Since we use linear basis functions, according to the theory of finite element methods the error should be of order h^2. However, the use of iterative methods, timestepping and possibly stabilization methods might have a negative impact on the error.

- In the numerical experiments from Section 5.4 we mostly use parameters that are close to unity. For real world applications realistic values of these parameters should be chosen. However, correct values

are often unobtainable, in which case the tumor growth model can only be used for approximate, qualitative predictions.

- A more realistic model can be constructed by considering the three-dimensional equations, which means a three-dimensional mesh has to be used. An intermediate step would be to consider a 'pseudo' three-dimensional model, in which we make use of rotational symmetry. In this case the model equations have to be adapted.

- In Appendix B we use delta functions to exactly invert the Laplace matrix in one dimension. A similar approach is used to approximate the inverse of the discrete Reaction-Laplacian operator in one dimension. We are interested to see if this method of finding (approximate) inverses is also applicable in two dimensions. It is expected to be more difficult because the Green's function of the Laplacian contains a singularity.

6.2 Suggestions for Improving Model Performance

The current implementation should be regarded as a proof of concept rather than a well-optimized numerical model. The simulations performed in Section 5.4 take only a few minutes, but the meshes are quite coarse. If the model is to be converted to three dimensions, the systems will grow from size $\mathcal{O}(n^2)$ to $\mathcal{O}(n^3)$, where n is the number of meshpoints. Using the current implementation the conversion from two to three dimensions will not be computationally feasible. There are numerous ways to improve its performance. We list a number of suggestions.

- To get more accurate results for growing domains remeshing could be applied. In the current implementation elements are deformed as a result of moving grid nodes. By generating a new, higher quality mesh every few timesteps the accuracy can be improved. The current mesh generating library, dmsh, is rather slow. Thus, for the purpose of remeshing it should be replaced. A possible candidate is the library MeshPy, which provides a Python interface to the two-dimensional mesh generator Triangle. This package less user-friendly but much faster. Another downside of dmsh is that it does not support three-dimensional meshes.

- To improve the computation time the model could be translated to C++ code. The most drastic speed-up will be seen in the matrix construction step. In the current model, all matrices are constructed by looping over every element and adding the corresponding contributions to the relevant matrix entries. Loops in Python are known to be slow compared to C or C++ loops. We suspect that the speed-up in the solving step is much smaller, because the libraries NumPy and SciPy already use C code 'under the hood'.

- Another advantage of C++ is the ability to easily adapt the code to run on a GPU. Both the matrix construction step and (sparse) system solving step can be performed in parallel, potentially leading to drastic speed-ups.

Bibliography

[1] G. Aguilar, F. Gaspar, F. Lisbona, and C. Rodrigo, *Numerical stabilization of biot's consolidation model by a perturbation on the flow equation*, International journal for numerical methods in engineering, 75 (2008), pp. 1282–1300.

[2] J. Bell, *Sobolev spaces in one dimension and absolutely continuous functions*, 2015.

[3] M. Bukac, I. Yotov, R. Zakerzadeh, and P. Zunino, *Effects of Poroelasticity on Fluid-Structure Interaction in Arteries: a Computational Sensitivity Study*, Springer International Publishing, Cham, 2015, pp. 197–220.

[4] R. M. Corless, G. H. Gonnet, D. E. Hare, D. J. Jeffrey, and D. E. Knuth, *On the lambert w function*, Advances in Computational mathematics, 5 (1996), pp. 329–359.

[5] S. C. Cowin, *Bone poroelasticity*, Journal of Biomechanics, 32 (1999), pp. 217 – 238.

[6] G. Dziuk and C. M. Elliott, *Finite elements on evolving surfaces*, IMA Journal of Numerical Analysis, 27 (2007), pp. 262–292.

[7] M. Fraldi and A. R. Carotenuto, *Cells competition in tumor growth poroelasticity*, Journal of the Mechanics and Physics of Solids, 112 (2018), pp. 345 – 367.

[8] C. L. Hall, *Modelling of some biological materials using continuum mechanics*, PhD thesis, Queensland University of Technology, 2008.

[9] R. Jia, *MATH 418: Honors Real Variable II Lecture Notes*, University of Alberta, 2019, ch. 3: Absolutely Continuous Functions.

[10] E. Kleimann, *Mathematical modelling of burn injuries*, Master's thesis, Delft University of Technology, 10 2018. Supervisor: Fred Vermolen.

[11] D. Koppenol, *Biomedical implications from mathematical models for the simulation of dermal wound healing*, PhD thesis, Delft University of Technology, The address of the publisher, 6 2017.

[12] E. H. Lee, *Elastic-plastic deformation at finite strains*, Journal of Applied Mechanics, (1969).

[13] C. Rodrigo, F. Gaspar, X. Hu, and L. T. Zikatanov, *Stability and monotonicity for some discretizations of the biot's consolidation model*, Computer Methods in Applied Mechanics and Engineering, 298 (2016), pp. 183–204.

[14] T. Roose, P. A. Netti, L. L. Munn, Y. Boucher, and R. K. Jain, *Solid stress generated by spheroid growth estimated using a linear poroelasticity model*, Microvascular research, 66 (2003), pp. 204–212.

[15] N. Schlömer, *dmsh github*. `https://github.com/nschloe/dmsh`. Accessed: 3/7/2020.

[16] D. Smits, *Morphoelastic models for burn contraction*, Master's thesis, Delft University of Technology, 9 2019. Supervisor: Fred Vermolen.

[17] F. Vermolen and A. Segal, *On an integration rule for products of barycentric coordinates of simplexes in* $\mathbb{R}^n$, Journal of Computational and Applied Mathematics, 330 (2018), pp. 289–294.

[18] C. Vuik, F. Vermolen, M. van Gijzen, and M. Vuik, *Numerical Methods for Ordinary Differential Equations*, Delft Academic Press, 2 ed., 2016.

[19] R. A. Weinberg and R. A. Weinberg, *The biology of cancer*, Garland science, 2013.

A

Element Integrals

In this chapter, various integrals that appear in the Galerkin equations are computed over a triangular element. We consider the element T with vertices $\boldsymbol{x}_1, \boldsymbol{x}_2$ and $\boldsymbol{x}_3$. We use linear basis functions and triangular elements, and write

$$\varphi_i(x, y) = \alpha_i x + \beta_i y + \gamma_i. \tag{A.1}$$

for $i \in \{1, 2, 3\}$. Thus, whenever the partial derivatives of a basis function appear, we substitute

$$\frac{\partial \varphi_i}{\partial x} = \alpha_i, \quad \text{and} \quad \frac{\partial \varphi_i}{\partial y} = \beta_i. \tag{A.2}$$

The α_i and β_i can be computed using the relation $\varphi_i(\boldsymbol{x}_j) = 1$, leading to the linear system

$$\begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix} \begin{pmatrix} \alpha_i \\ \beta_i \\ \gamma_i \end{pmatrix} = \boldsymbol{e}_i, \tag{A.3}$$

where $\boldsymbol{e}_i$ contains a 1 in its i'th position and zeroes otherwise. The area of the triangle T can also be expressed in terms of the matrix above:

$$|T| = \frac{1}{2} \det \begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix}. \tag{A.4}$$

This follows from a well-known theorem by Holand & Bell, see for example [17]. In this chapter we will give the element matrices and vectors corresponding to matrices and vectors found in the main text of this work. For example, let us consider a mass matrix, often defined as

$$M_{ij} = \int_\Omega \varphi_i \varphi_j \, \mathrm{d}\Omega. \tag{A.5}$$

To construct this matrix, we look at the contribution each element in the triangulation makes per entry:

$$M_{ij} = \sum_{T \in \mathcal{T}} \int_T \varphi_i \varphi_j \, \mathrm{d}\Omega. \tag{A.6}$$

Note that we can also view the summand in the right-hand side as an $n \times n$ matrix. However, since the linear basis functions φ_i and φ_j are only non-zero on elements containing grid node i or j respectively, this matrix will have 9 non-zero entries. These entries correspond to the integrals in which both grid node i and j are vertices of T. When constructing M, it is therefore sufficient to compute for each element T the 3×3 matrix

$$(M_T)_{ij} = \int_T \varphi_{i,T} \varphi_{j,T} \, \mathrm{d}\Omega, \tag{A.7}$$

for $i, j \in \{1, 2, 3\}$. Here we use (i, T) to denote the i'th vertex of T. In the following sections, we simply write i instead of (i, T), since it is clear from the context that only one element and its three vertices are considered.

A.1 Pure Elasticity

We give the element matrices and vectors corresponding to the objects found in system (2.25) from Section 2.2.2. The 3×3 element block matrices $S_T^{11}, S_T^{12}, S_T^{21}$ and S_T^{22} are given by

$$(S_T^{11})_{ij} = \int_T (\mu+\lambda)\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + \frac{1}{2}\mu\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y}\,\mathrm{d}\Omega = |T|\left((\mu+\lambda)\alpha_i\alpha_j + \frac{1}{2}\mu\beta_i\beta_j\right), \tag{A.8}$$

$$(S_T^{12})_{ij} = \int_T \lambda\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \frac{1}{2}\mu\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x}\,\mathrm{d}\Omega = |T|\left(\lambda\alpha_i\beta_j + \frac{1}{2}\mu\beta_i\alpha_j\right), \tag{A.9}$$

$$(S_T^{21})_{ij} = \int_T \frac{1}{2}\mu\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial y} + \lambda\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial x}\,\mathrm{d}\Omega = |T|\left(\frac{1}{2}\mu\alpha_i\beta_j + \lambda\beta_i\alpha_j\right), \tag{A.10}$$

$$(S_T^{22})_{ij} = \int_T \frac{1}{2}\mu\frac{\partial\varphi_i}{\partial x}\frac{\partial\varphi_j}{\partial x} + (\mu+\lambda)\frac{\partial\varphi_i}{\partial y}\frac{\partial\varphi_j}{\partial y}\,\mathrm{d}\Omega = |T|\left(\frac{1}{2}\mu\alpha_i\alpha_j + (\mu+\lambda)\beta_i\beta_j\right), \tag{A.11}$$

Using the outer product of two vectors, the element matrices can be written in the following form:

$$S_T^{11} = |T|\left((\mu+\lambda)\boldsymbol{\alpha}\boldsymbol{\alpha}^\top + \frac{1}{2}\mu\boldsymbol{\beta}\boldsymbol{\beta}^\top\right), \tag{A.12}$$

$$S_T^{12} = |T|\left(\lambda\boldsymbol{\alpha}\boldsymbol{\beta}^\top + \frac{1}{2}\mu\boldsymbol{\beta}\boldsymbol{\alpha}^\top\right), \tag{A.13}$$

$$S_T^{21} = |T|\left(\frac{1}{2}\mu\boldsymbol{\alpha}\boldsymbol{\beta}^\top + \lambda\boldsymbol{\beta}\boldsymbol{\alpha}^\top\right), \tag{A.14}$$

$$S_T^{22} = |T|\left(\frac{1}{2}\mu\boldsymbol{\alpha}\boldsymbol{\alpha}^\top + (\mu+\lambda)\boldsymbol{\beta}\boldsymbol{\beta}^\top\right). \tag{A.15}$$

Where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the column vectors

$$\boldsymbol{\alpha} = \begin{bmatrix}\alpha_1\\ \alpha_2\\ \alpha_3\end{bmatrix}, \qquad \boldsymbol{\beta} = \begin{bmatrix}\beta_1\\ \beta_2\\ \beta_3\end{bmatrix}. \tag{A.16}$$

Expressions (A.8)-(A.11) can be used to construct (2.30)-(2.33) by summing over all elements. The entries of the right-hand side vectors $\boldsymbol{g}$ and $\boldsymbol{\tau}$ can be approximated on the triangle T using a first order Newton-Cotes rule:

$$(g_T^1)_i = \int_T \varphi_i g^1\,\mathrm{d}\Omega \approx \frac{|T|}{3}g^1(\boldsymbol{x}_i), \qquad i = 1,2,3. \tag{A.17}$$

And similarly for $(g_T^2)_i$. Let T_B be the boundary element (line) connecting the mesh points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. The boundary element vector component $(\tau_{T_B}^1)_i$ is approximated by

$$(\tau_{T_B}^1)_i = \int_{T_B} \varphi_i\tau^1\,\mathrm{d}\Gamma \approx \frac{|T_B|}{2}\tau^1(\boldsymbol{x}_i), \qquad i = 1,2. \tag{A.18}$$

Where $|T_B|$ is the length of T_B. The complete vector can be found by summing the (boundary) element vector components over all elements and boundary elements respectively. For example, we have

$$(g^1)_i = \sum_{T\in\mathcal{T}}(g_T^1)_i, \qquad \text{and} \qquad (\tau^1)_i = \sum_{T_B\in\mathcal{T}_B}(\tau_{T_B}^1)_i, \tag{A.19}$$

where $\mathcal{T}$ denotes a triangulation of Ω and $\mathcal{T}_B$ is the corresponding partition of Γ_1.

A.2 Morphoelasticity

We give all element matrices and vectors relative to T of the objects that make up the system in (2.109) and (2.119). First, consider the mass matrix:

$$M_T = \frac{|T|}{12}\begin{bmatrix}2 & 1 & 1\\ 1 & 2 & 1\\ 1 & 1 & 2\end{bmatrix}, \tag{A.20}$$

Next, we consider the blocks that make up the viscous stress matrix. Note that the elastic stress matrix S_{el} found in sections 2.2 and 2.3 is very similar. Compared to the the following expressions, the parameters μ_v, λ_v should be replaced by μ_ε, λ_ε.

$$(S^{11}_{\text{vis}})_T = |T|(\mu_v + \lambda_v)\begin{bmatrix}\alpha_1\alpha_1 & \alpha_1\alpha_2 & \alpha_1\alpha_3\\ \alpha_2\alpha_1 & \alpha_2\alpha_2 & \alpha_2\alpha_3\\ \alpha_3\alpha_1 & \alpha_3\alpha_2 & \alpha_3\alpha_3\end{bmatrix} + \frac{1}{2}|T|\mu_v\begin{bmatrix}\beta_1\beta_1 & \beta_1\beta_2 & \beta_1\beta_3\\ \beta_2\beta_1 & \beta_2\beta_2 & \beta_2\beta_3\\ \beta_3\beta_1 & \beta_3\beta_2 & \beta_3\beta_3\end{bmatrix}, \tag{A.21}$$

$$(S^{12}_{\text{vis}})_T = |T|\lambda_v\begin{bmatrix}\alpha_1\beta_1 & \alpha_1\beta_2 & \alpha_1\beta_3\\ \alpha_2\beta_1 & \alpha_2\beta_2 & \alpha_2\beta_3\\ \alpha_3\beta_1 & \alpha_3\beta_2 & \alpha_3\beta_3\end{bmatrix} + \frac{1}{2}|T|\mu_v\begin{bmatrix}\beta_1\alpha_1 & \beta_1\alpha_2 & \beta_1\alpha_3\\ \beta_2\alpha_1 & \beta_2\alpha_2 & \beta_2\alpha_3\\ \beta_3\alpha_1 & \beta_3\alpha_2 & \beta_3\alpha_3\end{bmatrix}, \tag{A.22}$$

$$(S^{21}_{\text{vis}})_T = \frac{1}{2}|T|\mu_v\begin{bmatrix}\alpha_1\beta_1 & \alpha_1\beta_2 & \alpha_1\beta_3\\ \alpha_2\beta_1 & \alpha_2\beta_2 & \alpha_2\beta_3\\ \alpha_3\beta_1 & \alpha_3\beta_2 & \alpha_3\beta_3\end{bmatrix} + |T|\lambda_v\begin{bmatrix}\beta_1\alpha_1 & \beta_1\alpha_2 & \beta_1\alpha_3\\ \beta_2\alpha_1 & \beta_2\alpha_2 & \beta_2\alpha_3\\ \beta_3\alpha_1 & \beta_3\alpha_2 & \beta_3\alpha_3\end{bmatrix}, \tag{A.23}$$

$$(S^{22}_{\text{vis}})_T = \frac{1}{2}|T|\mu_v\begin{bmatrix}\alpha_1\alpha_1 & \alpha_1\alpha_2 & \alpha_1\alpha_3\\ \alpha_2\alpha_1 & \alpha_2\alpha_2 & \alpha_2\alpha_3\\ \alpha_3\alpha_1 & \alpha_3\alpha_2 & \alpha_3\alpha_3\end{bmatrix} + |T|(\lambda_v + \mu_v)\begin{bmatrix}\beta_1\beta_1 & \beta_1\beta_2 & \beta_1\beta_3\\ \beta_2\beta_1 & \beta_2\beta_2 & \beta_2\beta_3\\ \beta_3\beta_1 & \beta_3\beta_2 & \beta_3\beta_3\end{bmatrix}, \tag{A.24}$$

The block matrices B_x and B_y frequently occur in the system, its element matrices are given by:

$$(B_x)_T = \frac{|T|}{3}\begin{bmatrix}\alpha_1 & \alpha_1 & \alpha_1\\ \alpha_2 & \alpha_2 & \alpha_2\\ \alpha_3 & \alpha_3 & \alpha_3\end{bmatrix}, \quad (B_y)_T = \frac{|T|}{3}\begin{bmatrix}\beta_1 & \beta_1 & \beta_1\\ \beta_2 & \beta_2 & \beta_2\\ \beta_3 & \beta_3 & \beta_3\end{bmatrix}. \tag{A.25}$$

Furthermore, the element vectors are given by:

$$(g^1)_T = \frac{|T|}{3}\begin{bmatrix}g^1(\boldsymbol{x}_1)\\ g^1(\boldsymbol{x}_2)\\ g^1(\boldsymbol{x}_3)\end{bmatrix}, \quad (g^2)_T = \frac{|T|}{3}\begin{bmatrix}g^2(\boldsymbol{x}_1)\\ g^2(\boldsymbol{x}_2)\\ g^2(\boldsymbol{x}_3)\end{bmatrix}, \tag{A.26}$$

$$(G^{11})_T = \frac{|T|}{3}\begin{bmatrix}G^{11}(\boldsymbol{x}_1)\\ G^{11}(\boldsymbol{x}_2)\\ G^{11}(\boldsymbol{x}_3)\end{bmatrix}, \quad (G^{21})_T = \frac{|T|}{3}\begin{bmatrix}G^{12}(\boldsymbol{x}_1)\\ G^{12}(\boldsymbol{x}_2)\\ G^{12}(\boldsymbol{x}_3)\end{bmatrix}, \quad (G^{22})_T = \frac{|T|}{3}\begin{bmatrix}G^{22}(\boldsymbol{x}_1)\\ G^{22}(\boldsymbol{x}_2)\\ G^{22}(\boldsymbol{x}_3)\end{bmatrix}. \tag{A.27}$$

Let us now consider the element matrices involving the non-linear part of (2.109). Note that we can view $(N^x)_T$ and $(N^y)_T$ as $3\times 3\times 3$ tensors that we can write as

$$((N^x)_T)_{ijk} = \alpha_k\frac{|T|}{3}(1+\delta_{ij}), \quad ((N^y)_T)_{ijk} = \beta_k\frac{|T|}{3}(1+\delta_{ij}). \tag{A.28}$$

Thus, we get the following element matrices:

$$((N^x)_T)_{1\cdot\cdot} = \frac{|T|}{3}\begin{bmatrix}2\alpha_1 & 2\alpha_2 & 2\alpha_3\\ \alpha_1 & \alpha_2 & \alpha_3\\ \alpha_1 & \alpha_2 & \alpha_3\end{bmatrix}, \tag{A.29}$$

$$((N^x)_T)_{2\cdot\cdot} = \frac{|T|}{3}\begin{bmatrix}\alpha_1 & \alpha_2 & \alpha_3\\ 2\alpha_1 & 2\alpha_2 & 2\alpha_3\\ \alpha_1 & \alpha_2 & \alpha_3\end{bmatrix}, \tag{A.30}$$

$$((N^x)_T)_{3\cdot\cdot} = \frac{|T|}{3}\begin{bmatrix}\alpha_1 & \alpha_2 & \alpha_3\\ \alpha_1 & \alpha_2 & \alpha_3\\ 2\alpha_1 & 2\alpha_2 & 2\alpha_3\end{bmatrix}, \tag{A.31}$$

and

$$((N^y)_T)_{1\cdot\cdot} = \frac{|T|}{3}\begin{bmatrix}2\beta_1 & 2\beta_2 & 2\beta_3\\ \beta_1 & \beta_2 & \beta_3\\ \beta_1 & \beta_2 & \beta_3\end{bmatrix}, \tag{A.32}$$

$$((N^y)_T)_{2\cdot\cdot} = \frac{|T|}{3}\begin{bmatrix}\beta_1 & \beta_2 & \beta_3\\ 2\beta_1 & 2\beta_2 & 2\beta_3\\ \beta_1 & \beta_2 & \beta_3\end{bmatrix}, \tag{A.33}$$

$$((N^y)_T)_{3\cdot\cdot} = \frac{|T|}{3}\begin{bmatrix}\beta_1 & \beta_2 & \beta_3\\ \beta_1 & \beta_2 & \beta_3\\ 2\beta_1 & 2\beta_2 & 2\beta_3\end{bmatrix}. \tag{A.34}$$

Finally, let T_B be the boundary element connecting vertices $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, then:

$$(\tau^1)_{T_B} = \frac{|T_B|}{2}\begin{bmatrix}\tau^1(\boldsymbol{x}_1)\\ \tau^1(\boldsymbol{x}_2)\end{bmatrix}, \quad (\tau^2)_{T_B} = \frac{|T_B|}{2}\begin{bmatrix}\tau^2(\boldsymbol{x}_1)\\ \tau^2(\boldsymbol{x}_2)\end{bmatrix}, \tag{A.35}$$

where $|T_B|$ denotes the length of T_B.

A.3 Bubble Functions

We give the element matrices and vector corresponding to bubble matrices S_b, D_b and bubble vector $\boldsymbol{g}_b$ from Section 3.3.2. Let $\varphi_{b,k}$ be the bubble function corresponding to element T_k, where $k \in \{1, \dots, n_T\}$.

$$(S_b^{11})_{kk} = (\mu+\lambda)\int_{T_k} \frac{\partial\varphi_{b,k}}{\partial x}\frac{\partial\varphi_{b,k}}{\partial x}\,\mathrm{d}\Omega + \frac{1}{2}\mu\int_{T_k}\frac{\partial\varphi_{b,k}}{\partial y}\frac{\partial\varphi_{b,k}}{\partial y}\,\mathrm{d}\Omega \tag{A.36}$$

$$(S_b^{12})_{kk} = \lambda\int_{T_k} \frac{\partial\varphi_{b,k}}{\partial x}\frac{\partial\varphi_{b,k}}{\partial y}\,\mathrm{d}\Omega + \frac{1}{2}\mu\int_{T_k}\frac{\partial\varphi_{b,k}}{\partial y}\frac{\partial\varphi_{b,k}}{\partial x}\,\mathrm{d}\Omega \tag{A.37}$$

$$(S_b^{21})_{kk} = \frac{1}{2}\mu\int_{T_k} \frac{\partial\varphi_{b,k}}{\partial x}\frac{\partial\varphi_{b,k}}{\partial y}\,\mathrm{d}\Omega + \lambda\int_{T_k}\frac{\partial\varphi_{b,k}}{\partial y}\frac{\partial\varphi_{b,k}}{\partial x}\,\mathrm{d}\Omega \tag{A.38}$$

$$(S_b^{22})_{kk} = \frac{1}{2}\mu\int_{T_k} \frac{\partial\varphi_{b,k}}{\partial x}\frac{\partial\varphi_{b,k}}{\partial x}\,\mathrm{d}\Omega + (\mu+\lambda)\int_{T_k}\frac{\partial\varphi_{b,k}}{\partial y}\frac{\partial\varphi_{b,k}}{\partial y}\,\mathrm{d}\Omega \tag{A.39}$$

Note that all the S_b blocks are diagonal matrices, because a bubble-function is only non-zero on its corresponding element. This is also why can integrate over T_k instead of Ω_h. The three distinct integrals in the above formulas are given by:

$$\int_{T_k} \frac{\partial\varphi_{b,k}}{\partial x}\frac{\partial\varphi_{b,k}}{\partial x}\,\mathrm{d}\Omega = \frac{81|T_k|}{10}\left[\alpha_1^2+\alpha_1\alpha_2+\alpha_2^2+\alpha_2\alpha_3+\alpha_3^2+\alpha_1\alpha_3\right] \tag{A.40}$$

$$\begin{aligned}\int_{T_k} \frac{\partial\varphi_{b,k}}{\partial x}\frac{\partial\varphi_{b,k}}{\partial y}\,\mathrm{d}\Omega = \frac{81|T_k|}{20}[&2\alpha_1\beta_1+\alpha_1\beta_2+\alpha_1\beta_3\\ &+\alpha_2\beta_1+2\alpha_2\beta_2+\alpha_2\beta_3+\alpha_3\beta_1+\alpha_3\beta_2+2\alpha_3\beta_3]\end{aligned} \tag{A.41}$$

$$\int_{T_k} \frac{\partial\varphi_{b,k}}{\partial y}\frac{\partial\varphi_{b,k}}{\partial y}\,\mathrm{d}\Omega = \frac{81|T_k|}{10}\left[\beta_1^2+\beta_1\beta_2+\beta_2^2+\beta_2\beta_3+\beta_3^2+\beta_1\beta_3\right] \tag{A.42}$$

The matrix D_b is also subdivided into blocks: $D_b = \begin{pmatrix}D_b^1 & D_b^2\end{pmatrix}$. They are given component-wise by:

$$(D_b^1)_{kj} = \int_{\Omega_h}\frac{\partial\varphi_{b,k}}{\partial x}\psi_j\,\mathrm{d}\Omega, \qquad (D_b^2)_{kj} = \int_{\Omega_h}\frac{\partial\varphi_{b,k}}{\partial y}\psi_j\,\mathrm{d}\Omega, \tag{A.43}$$

where $k \in \{1,\dots,n_T\}$ and $j \in \{1,\dots,n\}$ and ψ_j is a linear basis function. The corresponding element matrices are given by:

$$(D_b^1)_T = \frac{9|T|}{20}\begin{bmatrix}\alpha_1+2\alpha_2+2\alpha_3\\ 2\alpha_1+\alpha_2+2\alpha_3\\ 2\alpha_1+2\alpha_2+\alpha_3\end{bmatrix}, \qquad (D_b^2)_T = \frac{9|T|}{20}\begin{bmatrix}\beta_1+2\beta_2+2\beta_3\\ 2\beta_1+\beta_2+2\beta_3\\ 2\beta_1+2\beta_2+\beta_3\end{bmatrix}. \tag{A.44}$$

We also write $\boldsymbol{g}_b^m = ((\boldsymbol{g}_b^1)^m, (\boldsymbol{g}_b^2)^m)^\top$, they are given by:

$$(\boldsymbol{g}_b^1)_{T_k}^m = \int_{T_k}\varphi_{b,k}(g^1)^m\,\mathrm{d}\Omega, \qquad (\boldsymbol{g}_b^1)_{T_k}^m = \int_{T_k}\varphi_{b,k}(g^2)^m\,\mathrm{d}\Omega. \tag{A.45}$$

The integral is approximated using Newton-Cotes quadrature, where we use the barycentric coordinates of T_k to linearly approximate g^1 and g^2. Holand & Bell's theorem can then be used to exactly compute the resulting integrals, such that we obtain:

$$(\boldsymbol{g}_b^1)_{T_k}^m \approx \frac{3|T_k|}{20}\left(g^1(\boldsymbol{x}_1,t^m)+g^1(\boldsymbol{x}_2,t^m)+g^1(\boldsymbol{x}_3,t^m)\right), \tag{A.46}$$

$$(\boldsymbol{g}_b^2)_{T_k}^m \approx \frac{3|T_k|}{20}\left(g^2(\boldsymbol{x}_1,t^m)+g^2(\boldsymbol{x}_2,t^m)+g^2(\boldsymbol{x}_3,t^m)\right). \tag{A.47}$$

A.4 Tumor Growth Model

Although most integrals found in the tumor growth model have already been calculated in previous sections, a handful of new ones are introduced. Note that the last integral in (5.56) is a kind of mass matrix with an additional basis function, we denote it by:

$$((\mathcal{M}_\ell)_T)_{ij} = \int_T \varphi_{\ell,T}\varphi_i\varphi_j \, \mathrm{d}\Omega. \tag{A.48}$$

Its element matrices are given by:

$$(\mathcal{M}_1)_T = \frac{|T|}{60}\begin{bmatrix}6&2&2\\2&2&1\\2&1&2\end{bmatrix}, \quad (\mathcal{M}_2)_T = \frac{|T|}{60}\begin{bmatrix}2&2&1\\2&6&2\\1&2&2\end{bmatrix}, \quad (\mathcal{M}_3)_T = \frac{|T|}{60}\begin{bmatrix}2&1&2\\1&2&2\\2&2&6\end{bmatrix}. \tag{A.49}$$

The above integral also occurs in (5.66). The other integral in this equation is similar, but contains yet another basis function. We denote it by

$$((\mathcal{M}_{k\ell})_T)_{ij} = \int_T \varphi_i\varphi_j\varphi_{k,T}\varphi_{\ell,T} \, \mathrm{d}\Omega. \tag{A.50}$$

Its nine corresponding element matrices are given by

$$(\mathcal{M}_{11})_T = \frac{|T|}{180}\begin{bmatrix}12&3&3\\3&2&1\\3&1&2\end{bmatrix} \qquad (\mathcal{M}_{12})_T = \frac{|T|}{180}\begin{bmatrix}3&2&1\\2&3&1\\1&1&1\end{bmatrix} \qquad (\mathcal{M}_{13})_T = \frac{|T|}{180}\begin{bmatrix}3&1&2\\1&1&1\\2&1&3\end{bmatrix}$$

$$(\mathcal{M}_{21})_T = \frac{|T|}{180}\begin{bmatrix}3&2&1\\2&3&1\\1&1&1\end{bmatrix} \qquad (\mathcal{M}_{22})_T = \frac{|T|}{180}\begin{bmatrix}2&3&1\\3&12&3\\1&3&2\end{bmatrix} \qquad (\mathcal{M}_{23})_T = \frac{|T|}{180}\begin{bmatrix}1&1&1\\1&3&2\\1&2&3\end{bmatrix}$$

$$(\mathcal{M}_{31})_T = \frac{|T|}{180}\begin{bmatrix}3&1&2\\1&1&1\\2&1&3\end{bmatrix} \qquad (\mathcal{M}_{32})_T = \frac{|T|}{180}\begin{bmatrix}1&1&1\\1&3&2\\1&2&3\end{bmatrix} \qquad (\mathcal{M}_{33})_T = \frac{|T|}{180}\begin{bmatrix}2&1&3\\1&2&3\\3&3&12\end{bmatrix}$$

B

Matrix Inversion by Analyzing Delta Problems

B.1 Inverting the Laplace Matrix in 1D

In section 4.3.3 we computed an approximation to the inverse of a discrete differential operator. It turns out that for the Laplace operator, this approximation is exact. We analyze the following equation:

$$-u'' = \delta(x-a), \qquad x \in (0,1) \tag{B.1}$$

where $a \in (0,1)$. A number of different sets of boundary conditions can be imposed. We will first consider homogeneous Dirichlet boundary conditions:

$$u(0) = u(1) = 0. \tag{B.2}$$

Later we will also consider different sets of boundary conditions. The exact (weak) solution to (B.1)-(B.2) is given by:

$$u(x) = (1-a)x - (x-a)H(x-a), \tag{B.3}$$

where $H(\cdot)$ is the Heaviside step-function. The exact solution can be verified by integrating the differential equation twice, and substituting the boundary condtions to obtain the values of the integration constants. Another option would be to use the Laplace transform. Note that u is not a classical solution, but we do have $u \in H^1(0,1)$. We build a finite element model to approximate the solution to (B.1). First, consider the weak form:

Find $u \in H_0^1(0,1)$ such that

$$\int_0^1 u'\varphi' \,\mathrm{d}x = \varphi(a), \qquad \forall \varphi \in H_0^1(0,1), \tag{B.4}$$

where $H_0^1(0,1) = \{u \in H^1(0,1) : u(0) = u(1) = 0\}$.

Divide $(0,1)$ into a uniform mesh consisting of $n+1$ elements. Let $h = 1/(n+1)$ be the meshwidth and let $x_i = ih$ be a grid point for $i = 0, \ldots, n+1$. Note that since $u(0) = u(1) = 0$, the unknowns are located at $x_1, \ldots, x_n$. Let φ_i be a piecewice linear basis function such that $\varphi_i(x_j) = \delta_{ij}$, and let u_h be the finite element approximation to u. We write

$$u_h(x) = \sum_{j=1}^{n} c_j \varphi_j(x). \tag{B.5}$$

Substitute u_h into the weak form (B.4), and set $\varphi = \varphi_i$ to obtain

$$\sum_{j=1}^{n} c_j \int_0^1 \varphi_i' \varphi_j' \,\mathrm{d}x = \varphi_i(a), \qquad i = 1, \ldots, n. \tag{B.6}$$

We let a coincide with one of the grid points, say $a = x_k$ for some $k \in \{1, \dots, n\}$. Then (B.6) turns into:

$$\sum_{j=1}^{n} c_j \int_0^1 \varphi_i' \varphi_j' \, \mathrm{d}x = \delta_{ik}, \qquad i = 1, \dots, n. \tag{B.7}$$

Let

$$S_{ij} = \int_0^1 \varphi_i' \varphi_j' \, \mathrm{d}x, \tag{B.8}$$

then the matrix S is given by

$$S = \frac{1}{h} \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}. \tag{B.9}$$

Theorem 12. *Let the $n \times n$ matrix S be given by* (B.9)*, and let $\boldsymbol{e}_k$ be the k'th standard basis vector in $\mathbb{R}^n$, such that $e_k^j = \delta_{kj}$. Then the vector $\boldsymbol{v}_k \in \mathbb{R}^n$ given by*

$$v_k^j = (1 - x_k)x_j - (x_j - x_k)H(j - k) \tag{B.10}$$

solves the equation $S\boldsymbol{v}_k = \boldsymbol{e}_k$ for $k = 1, \dots, n$. Therefore, the entries of S^{-1} are defined by (B.10)*.*

Proof. We use Céa's Lemma to prove the theorem. Let u and u_h be the exact solution and finite element solution respectively. Assume that u_h satisfies (B.7). Then by Céa's Lemma we have

$$\|u' - u_h'\|_{L^2(0,1)} \leqslant \|u' - v'\|_{L^2(0,1)}$$

for all $v \in \operatorname{Span}\{\varphi_i\}_{i=1}^n$. Note that u is linear on each element. Thus, by choosing for v the linear interpolant of u, the error becomes zero. Then, by the Poincaré inequality we obtain

$$\|u - u_h\|_{L^2(0,1)} \leqslant c\|u' - u_h'\|_{L^2(0,1)} \leqslant 0.$$

Because the finite element error is zero, expression (B.3) can be used to construct the finite element solution. Thus, by setting $a = x_k$ in (B.3), we find that the solution to $S\boldsymbol{v}_k = \boldsymbol{e}_k$ is given by:

$$v_k^j = (1 - x_k)x_j - (x_j - x_k)H(j - k).$$

Consider the matrix equation

$$S \begin{pmatrix} \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_n \end{pmatrix} = \begin{pmatrix} \boldsymbol{e}_1 & \cdots & \boldsymbol{e}_n \end{pmatrix} = I,$$

then it is clear that

$$S^{-1} = \begin{pmatrix} \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_n \end{pmatrix}.$$

□

Since we have found an expression for the inverse of S, general systems of the form

$$S\boldsymbol{u} = \boldsymbol{f} \tag{B.11}$$

can now easily be solved by setting

$$\boldsymbol{u} = \sum_{k=1}^{n} f_k \boldsymbol{v}_k. \tag{B.12}$$

B.2 Connection to Green's Functions

It is well known that for any linear differential operator L, the problem

$$Lu(x) = f(x) \tag{B.13}$$

can be solved exactly by using the Green's function corresponding to L. The Green's function $G(x, s)$ satisfies the equation

$$LG(x, s) = \delta(x - s). \tag{B.14}$$

Then the solution to (B.13) is given by

$$u(x) = \int G(x,s)f(s)\,\mathrm{d}s. \tag{B.15}$$

This can be easily verified by noting that L is linear and only acts on the variable x, thus:

$$\begin{aligned} Lu(x) &= L\int G(x,s)f(s)\,\mathrm{d}s \\ &= \int LG(x,s)f(s)\,\mathrm{d}s \\ &= \int \delta(x-s)f(x)\,\mathrm{d}s \\ &= f(x). \end{aligned}$$

Note that in our case $L = -\frac{\mathrm{d}^2}{\mathrm{d}x^2}$ with boundary conditions (B.2). The corresponding Green's function is given by (B.3). We rewrite it to fit the notation of this section:

$$G(x,s) = (1-s)x - (x-s)H(x-s). \tag{B.16}$$

By the previous section, we know that $G(x,x_k)$ is approximated exactly in a grid point x_k by the solution to $S\boldsymbol{v}_k = \boldsymbol{e}_k$, where S is given in (B.9). This fact can be used to prove that the finite element scheme, as introduced in the previous section, also approximates the solution to arbitrary problems of the form $-u''(x) = f(x)$ exactly.

Theorem 13. *Assume that u satisfies $-u''(x) = f(x)$ and the boundary conditions* (B.2)*. Let u_h be the corresponding finite element solution using linear basis functions on a uniform grid of $n+1$ elements. We write*

$$u_h(x) = \sum_{j=1}^{n} u_j\varphi_j(x). \tag{B.17}$$

Let the vector $\boldsymbol{u}$ contain the coefficients u_j. Then it satisfies the equation $S\boldsymbol{u} = \boldsymbol{f}$, where the matrix S is given by (B.9) *and $\boldsymbol{f}$ is defined pointwise by*

$$f_i = \int_0^1 f(x)\varphi_i(x)\,\mathrm{d}x. \tag{B.18}$$

Then $u_h(x_i) = u(x_i)$ in each grid point x_i.

For general functions f, the integrals in the right-hand side of (B.18) are usually approximated using some sort of Newton-Cotes quadrature rule. For the proof to hold, we need the elements of $\boldsymbol{f}$ to be exact.

Proof. Since $\boldsymbol{u}$ satisfies $S\boldsymbol{u} = \boldsymbol{f}$, by (B.11) and (B.12) we can write

$$\boldsymbol{u} = \sum_{k=1}^{n} f_k\boldsymbol{v}_k.$$

It follows that

$$\begin{aligned} u_h(x_i) &= \sum_{k=1}^{n} f_k v_k^i \\ &= \sum_{k=1}^{n} G(x_i,x_k)f_k \\ &= \sum_{k=1}^{n} G(x_i,x_k)\int_0^1 f(x)\varphi_k(x)\,\mathrm{d}x \\ &= \int_0^1 f(x)\sum_{k=1}^{n} G(x_i,x_k)\varphi_k(x)\,\mathrm{d}x \end{aligned}$$

Note that

$$\sum_{k=1}^{n} G(x_i, x_k)\varphi_k(x)$$

is the linear interpolant of $x \mapsto G(x_i, x)$, which is itself linear on each element, as can be seen in (B.16). It therefore follows that

$$\sum_{k=1}^{n} G(x_i, x_k)\varphi_k(x) = G(x_i, x)$$

for all $x \in (0, 1)$. Thus:

$$u_h(x_i) = \int_0^1 f(x)G(x_i, x)\,\mathrm{d}x = u(x_i)$$

by the definition of the Green's function and identity (B.15). □

Note that we can compute the integrals in (B.18) exactly for many different types of functions. For example: polynomials, piecewise polynomials, exponential functions, trigonometric function, et cetera. For all these types of source functions, the finite element approximation is exact.

B.3 Generalization to Non-Uniform Mesh

Note that the proof of Theorem 12 does not rely on the fact that the mesh is uniform. Consider a non-uniform mesh consisting of $n + 1$ elements. Let h_i be the length of the i'th element, and define the gridpoints by $x_i - x_{i-1} = h_i$, where $x_0 = 0$ and $x_{n+1} = 1$. Then the matrix S is now given by:

$$S = \begin{pmatrix} h_1^{-1} + h_2^{-1} & -h_2^{-1} & & & \\ -h_2^{-1} & h_2^{-1} + h_3^{-1} & -h_3^{-1} & & \\ & \ddots & \ddots & \ddots & \\ & & -h_{n-1}^{-1} & h_{n-1}^{-1} + h_n^{-1} & -h_n^{-1} \\ & & & -h_n^{-1} & h_n^{-1} + h_{n+1}^{-1} \end{pmatrix} \tag{B.19}$$

The entries of the inverse of S are again given by (B.10).

B.4 Higher Order Problems

The theory can easily be extended to higher order problems. For example, we can solve

$$S^2\boldsymbol{u} = \boldsymbol{f} \tag{B.20}$$

by solving subsequently

$$S\boldsymbol{u}_1 = \boldsymbol{f}, \quad \text{and} \quad S\boldsymbol{u} = \boldsymbol{u}_1. \tag{B.21}$$

We then get

$$u^j = \sum_{k=1}^{n} u_1^k v_k^j = \sum_{k=1}^{n}\sum_{\ell=1}^{n} f^\ell v_\ell^k v_k^j. \tag{B.22}$$

More generally, we can solve

$$S^p\boldsymbol{u} = \boldsymbol{f} \tag{B.23}$$

by setting

$$\begin{aligned} S\boldsymbol{u}_1 &= \boldsymbol{f} \\ S\boldsymbol{u}_2 &= \boldsymbol{u}_1 \\ &\vdots \\ S\boldsymbol{u}_p &= \boldsymbol{u}_{p-1}, \end{aligned}$$

and $\boldsymbol{u} = \boldsymbol{u}_p$. We then get the following cascading solution:

$$\begin{aligned} u^i &= \sum_{\alpha_{p-1}} u_{p-1}^{\alpha_{p-1}} v_{\alpha_{p-1}}^i \\ &= \sum_{\alpha_{p-2}} \sum_{\alpha_{p-1}} u_{p-2}^{\alpha_{p-2}} v_{\alpha_{p-2}}^{\alpha_{p-1}} v_{\alpha_{p-1}}^i \\ &\vdots \\ &= \sum_{\alpha_0} \cdots \sum_{\alpha_{p-1}} f^{\alpha_0} v_{\alpha_0}^{\alpha_1} v_{\alpha_1}^{\alpha_2} \cdots v_{\alpha_{p-2}}^{\alpha_{p-1}} v_{\alpha_{p-1}}^i. \end{aligned} \tag{B.24}$$

From the above expression we can derive the inverse of S^p element-wise. Indeed, we must have

$$u^i = \sum_{j=1}^{n} (S^{-p})_{ij} f^j, \tag{B.25}$$

and thus we find

$$(S^{-p})_{ij} = \sum_{\alpha_1} \cdots \sum_{\alpha_{p-1}} v_j^{\alpha_1} v_{\alpha_1}^{\alpha_2} \cdots v_{\alpha_{p-2}}^{\alpha_{p-1}} v_{\alpha_{p-1}}^i. \tag{B.26}$$

By choosing the right boundary conditions and order of the basis functions, the systems $S^p \boldsymbol{u} = \boldsymbol{f}$ correspond to finite element approximations of higher order problems $-u^{(p)} = f$. We will not investigate these connections any further at this time.

B.5 Mixed Boundary Conditions

To illustrate that we can also exactly invert other Laplace matrices, different boundary conditions are considered. We impose the following Robin or mixed boundary conditions:

$$-u'(0) + \alpha_0 u(0) = 0, \quad u'(1) + \alpha_1 u(1) = 0. \tag{B.27}$$

Integrating twice gives the following weak solution to this problem:

$$u(x) = c_0 + c_1 x - H(x-a)(x-a), \tag{B.28}$$

where the integration constants are given by

$$c_0 = \frac{\alpha_1(1-a)+1}{\alpha_0 + \alpha_0\alpha_1 + \alpha_1}, \quad c_1 = \frac{\alpha_0\alpha_1(1-a)+\alpha_0}{\alpha_0 + \alpha_0\alpha_1 + \alpha_1}.$$

Note that a solution only exists whenever $\alpha_0 + \alpha_0\alpha_1 + \alpha_1 \neq 0$. Due to the different choice of boundary conditions, the corresponding weak form is also different:

Find $u \in H^1(0,1)$ such that

$$\int_0^1 u'\varphi' \,\mathrm{d}x + \alpha_0 u(0)\varphi(0) + \alpha_1 u(1)\varphi(1) = \varphi(a), \qquad \forall \varphi \in H^1(0,1). \tag{B.29}$$

We consider again a uniform grid with $n+1$ and meshwidth $h = 1/(n+1)$. The unknowns are now located at $x_0, x_1, \ldots, x_{n+1}$, where $x_i = ih$. Also let a equal one of the grid points; $a = x_k$ for some $k \in \{0, \ldots, n+1\}$. Then the system of Galerkin equations is given by $S\boldsymbol{u}_k = \boldsymbol{e}_k$, where $\boldsymbol{e}_k$ is the k'th standard basis vector and S is the $(n+2) \times (n+2)$ matrix

$$S = \frac{1}{h} \begin{pmatrix} 1+\alpha_0 h & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1+\alpha_1 h \end{pmatrix}. \tag{B.30}$$

The proof of Theorem 12 can be repeated almost exactly to show that the inverse of S is given by the exact solution (B.28) evaluated at the grid points:

$$\begin{aligned} S^{-1}_{jk} &= c_0 + c_1 x_j - (x_j - x_k)H(j-k) \\ &= \frac{(\alpha_1(1-a)+1)(1+\alpha_0 x_j)}{\alpha_0 + \alpha_0\alpha_1 + \alpha_1} - (x_j - x_k)H(j-k). \end{aligned} \tag{B.31}$$