

Modeling of Blood Flow in Zebrafish With Cancer

Soran Najat Ibrahim

Master's Thesis, Spring 2020



This master's thesis is submitted under the master's programme *Computational Science*, with programme option *Mechanics*, at the Department of Mathematics, University of Oslo. The scope of the thesis is 60 credits.

The front page depicts a section of the root system of the exceptional Lie group E_8 , projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

Abstract

For decades, the improvement of cancer research has relied on in vivo representations for analyzing cancer development and treatment. Particularly, one of the main reasons why zebrafishes are suitable for studying a wide variety of cancer types is the dynamic visualization of tumor growth in vivo. We can observe tumor cells spreading locally into nearby healthy tissues, or globally through lymph and blood vessels. In the blood vessels, abnormally high permeability of the vessel walls can result in blood eruptions, consequently causing leakage of tumor cells into the surrounding tissue. In this thesis, we aim to model this phenomenon by means of partial differential equations. Different governing equations for flow modeling are proposed for different domains. Flow in the blood vessel (viscous domain) is modeled by simplified Navier-Stokes equations, while Darcy's law is representing the flow in the tissue (porous domain). Together, coupled Darcy-Stokes equations will be used as an approximation of the flow within and between the two domains. The advantage of this modeling is the possibility to observe eruptions that emerge at the interface (endothelial cells) between the domains. The results generated in this thesis illustrate the evolution of the eruption from various values of permeability in a simplified geometry. The accuracy of the simulated results have been verified by comparing them with available experimental data.

Acknowledgements

First of all, I would like to thank my supervisor Kent-Andre Mardal, for his guidance, knowledge, and the opportunity to work on this project. I would also like to thank my secondary supervisor Simon Wolfgang Funke for the guidance. Thanks to Petter Vollestad for the data and always have the doors open for questions. Thank you, Ana Budisa, for providing helpful comments and tips for the thesis. Finally, I would thank my friends and family for keeping me motivated throughout my five years.

List of Abbreviations and Notation

Boldface character denotes a vector. Moreover, the following list of abbreviations and symbols are used.

Abbreviations

FEM	Finite Element Method
PDE	Partial differential equation
CO ₂	Carbon dioxide
cP	Centipoise, measurement unit for viscosity
hpf	hours post-fertilization
dpf	days post-fertilization
$Pa \cdot s$	Pascal-second, measurement unit for dynamic viscosity
Re	Reynolds number
MMS	Method of Manufactured Solutions
UFL	Unified Form Language
VTK	Visualization Toolkit
PTV	Particle tracking velocimetry

Symbols

u	Velocity field
p	Pressure
γ	Shear rate
τ	Shear stress
ρ	Density
μ	Dynamic viscosity
i	Imaginary Number
Δp	Pressure gradient
κ	Flow consistency index
A	Cross section
Q	Volumetric flow rate
r	Radius
α	Womersley number, represented as : $\alpha = a\sqrt{\frac{n\omega}{\nu}}$
ν	Kinematic viscosity
ω	Angular frequency

J_β	Bessel functions of the first kind of order β
Y_β	Bessel functions of the second kind of order β
\mathbb{R}^n	Set of the n-tuples of real numbers
Ω	The domain of interest, an open subset of \mathbb{R}^n
Ω_s	Stokes part of the domain
Ω_d	Darcy part of the domain
$\sigma(\mathbf{u}, p)$	Cauchy stress tensor
$\epsilon(\cdot)$	Symmetric strain rate tensor
Γ	Interface between Ω_s and Ω_d
$\langle \cdot, \cdot \rangle$	L^2 -inner product
\mathbf{I}	Identity matrix
ϕ	Porosity, a measure of the void spaces in a material
n_s	Unit normal vector exterior to $\partial\Omega_s$
n_d	Unit normal vector exterior to $\partial\Omega_d$
τ	Unit tangential vector
K	Scalar permeability
\mathbf{K}	Tensor permeability
α	Beavers-Joseph-Saffmann coefficient
\forall	For all

List of Figures

2.1.1	Pulmonary and systemic circulation	5
2.1.2	Circulation in zebrafish	5
2.1.3	The fish heart	6
2.2.4	The Structure of Artery Wall	7
2.2.5	A Description of a slice of zebrafish	8
2.2.6	Eruption area one	8
2.2.7	Eruption area two	9
2.2.8	Eruption area three	9
2.2.9	Blood viscosity in zebrafish	10
2.2.10	Shear stress	11
2.2.11	Diameter	14
2.3.12	Different level of permeability	15
2.4.13	Laminar Eruption	16
2.4.14	Radial Eruption	16
2.4.15	3D laminar eruption	17
2.4.16	Diameter of the Eruption	17
2.4.17	3D Radial Eruption	18
3.2.1	An Example of fluid and porous zone	24
3.3.2	A View of Dirichlet boundaries	25
3.3.3	Simulation area	26
4.1.1	Taylor-Hood element in 2D	32
4.1.2	Taylor-Hood element in 3D	32
4.1.3	Crouzeix-Raviart element in 2D	33
4.2.4	Generated mesh for Poisson problem	38
4.2.5	Quick ParaView view over Poisson equation	38
4.3.6	Mesh converting	41
4.3.7	Simple 2D domain	42
4.3.8	ParaView	45
5.1.1	Length of the parts in the mesh	54
5.1.2	Element size	55
5.1.3	Meshes	56
5.1.4	Element size	57
5.3.5	Scenario 1: Low permeability	60
5.3.6	Scenario 1: Increased permeability	61
5.3.7	Scenario 1: Plot over line (Increased Permeability)	62
5.3.8	Scenario 1: Hyperpermeability	63
5.3.9	Scenario 1: Plot over line (Hyperpermeability)	63

5.3.10	Scenario 2: Low permeability	65
5.3.11	Scenario 2: Increased permeability	66
5.3.12	Scenario 2: Plot over line (Increased Permeability)	66
5.3.13	Scenario 2: Hyperpermeability	67
5.3.14	Scenario 2: Plot over line (Hyperpermeability)	68
5.4.15	Velocity development inside the simulated blood vessel	69
5.4.16	Simulated vs. Calculated extrusion velocity: Closed	70
5.4.17	Simulated vs. Calculated extrusion velocity: Open	71
5.4.18	Pressure change in the viscous domain	72
5.4.19	Pressure change in the closed grid	72
5.4.20	Pressure change in the open grid: Interface domain	73
5.4.21	Pressure change in the open grid: Porous domain	74

Contents

Acknowledgements	i
List of Abbreviations and Notation	i
List of Figures	iii
1 Introduction	1
2 Medical Background	4
2.1 The Circulatory System	4
2.1.1 Mammals	4
2.1.2 Zebrafish	5
2.1.3 Similarities Between Mammals and Fish	6
2.2 Flow in Blood Vessels	7
2.2.1 A Real Case of an Eruption	7
2.2.2 Viscosity	9
2.2.3 Poiseuille’s Equation and Laminar Flow	10
2.3 Permeability	14
2.4 Eruptions in Blood Vessels	15
2.4.1 Eruption	15
2.4.2 Laminar Eruption	16
2.4.3 Radial Eruption	18
3 Mathematical Model	21
3.1 Viscous Flow	21
3.1.1 The Navier-Stokes Equation	21
3.1.2 Stokes Equation	22
3.2 Porous Flow	22
3.2.1 Darcy’s Equation	23
3.3 Coupled Darcy–Stokes Flow	24
4 Numerical Methods	27
4.1 The Finite Element Method	27
4.1.1 Sobolev Spaces	28
4.1.2 Weak Formulation	29
4.1.3 Discretization	30
4.1.4 Finite Elements	31
4.2 FEniCS	34
4.2.1 Basic FEniCS Usage	34
4.2.2 The Complete Fenics Program	35
4.3 Advanced FEniCS	39
4.3.1 Mesh Generating Software	39

4.3.2	Geometry Storing	43
4.3.3	Dirichlet Boundaries	43
4.3.4	Mixed Elements	44
4.3.5	ParaView	44
4.4	Unified Discretizations of coupled Darcy–Stokes flow	45
4.4.1	The L^2 -Formulation	45
4.4.2	Implementation in FEniCS	50
5	Numerical Experiments	53
5.1	Element Scaling	53
5.1.1	Mesh	54
5.2	The Method of Manufactured Solutions	57
5.3	Numerical Experiments	58
5.3.1	Simulation Around the Interface	58
5.3.2	Simulation: Eruption in a Closed Grid	58
5.3.3	Simulation: Eruption in a Open Grid	64
5.4	Results	68
5.4.1	Velocity Development in the Coupled Scheme	68
5.4.2	Pressure Development in the Coupled Scheme	71
6	Summary and Discussion	75
6.1	Summary	75
6.1.1	The Coupled Darcy-Stokes Equation	75
6.1.2	The Domain	76
6.1.3	Boundaries	76
6.1.4	Simulations	76
6.1.5	Velocity and Pressure Development	77
6.2	Limitations	78
6.3	Further Work	78
6.4	Conclusion	79
A	Additional simulations	81

Chapter 1

Introduction

Cancer was recognized several thousand years ago, and is now one of the diseases which cause most deaths. The modern research and methods to study cancer have expanded since the end of the 19th-century [58]. A tumor manifests when the cells reproduce more than they should (or at an abnormal rate) without apparent purpose. Cancer is a particularly threatening type of tumor that has grown out-of-control (defined in [5]).

Zebrafish (*Danio rerio*) emerges as a powerful model to study human diseases and is serving as an alternative platform in cancer research. Zebrafish are used to study different varieties of tumors in nearly all organs, with features matching human tumors [24]. There are numerous methods to reproduce cancer in zebrafish, one of the main techniques are genetically manufacturing (reviewed in [61]). The biological features in zebrafish make it possible to study human cancer, for example, closed circulations system, visual blood at relatively low magnification, and the molecular mechanisms underlying vessel formation [16, 35]. The cancer cells spread locally by moving into nearby healthy tissues. On the global scale, the lymph and blood vessels are some of the transportation possibilities for the cancer cells to spread throughout the body [8]. Within the blood vessels, a single layer of endothelial cells constitutes the blood vessels walls, which controls the passage for cell-to-cell junction. Vascular permeability indicates the capacity in the intersections. Healthy tissues sustain vascular permeability at a low level, while increased vascular permeability results in plasma leakage, which will lead to various diseases [33, 45, 50]. If a blood vessel is transporting cancer cells, they may invade the blood vessel walls and increase vascular permeability. This can result in vigorous blood flow bursts, named eruptions, into the surrounding tissues.

Mathematical modeling is often used to simulate natural and medical phenomena. This can improve the understanding of complex processes and lead to

better engineering techniques. Simultaneous flow in both viscous and porous flow occurs in various fields. In geomechanics, applications such as groundwater flow across a soil or oil and gas flow in a permeable reservoir containing cracks. In biomechanics, the flows appear, for example, in the modeling of the interaction of blood flow surrounded by permeable tissues. The flow inside a blood vessel is represented as a viscous flow, while the covering tissues are represented as porous media. It is often easier to implement a numerical method for solving a coupled viscous and porous flow. Then the exact calculation which requires advanced experiments or images processing. The numerical methods are used to simulate medical procedures or complicated natural processes, which leads to an enhanced understanding of the processes and result in better engineering techniques. Stokes equations and Darcy's equation are used as the mathematical representation of the viscous and porous flow. This combination of flow is referred to as coupled Darcy-Stokes flow, connected with an interface.

In recent years, several methods for solving biomedical problems have been introduced. The papers by Dal et al. [7], Fullstone et al. [14], Gore et al. [16], Majno et al. [32], Matsumoto et al. [33] implies that the cause of endothelial cell contractions is increased permeability, which then supports the emigration of leukocytes and plasma proteins, or even the spread of diseases. One of the methods utilized for visualizing the events, is injecting nanoparticles into an appropriate area, and study the outcome with electron microscopy. In the paper by Dal et al. [7], a more sophisticated method has been presented for analyzing important parameters of nanoparticles behavior, especially interactions with the endothelial cells. However, in [14, 16, 32, 33], a standard injection method of nanoparticles are introduced. In the images by Resseguier [41, 42, 43, 44], nanoparticles are used for visualizing an inflammation. These images, together with the calculations performed by Vollestad [56, 57], will be compared with the simulations in the thesis.

One of the main approaches found in the literature to solve the coupled Darcy-Stokes equations is by the Finite Element Method (FEM) in either with the *decoupled* or *unified* strategy. The decoupled method consists of using one discretization for the Stokes equation and another one for the Darcy equation, which is connected through additional interface laws [46]. Conversely, a unified approach consists of solving the problem in the same finite element space. This method is introduced in the following papers [1, 3, 10, 11, 21]. The technique presented in this thesis is a steady-state unified strategy. We will consider a weak formulation (L^2 -formulation) of the Darcy-Stokes equations, which will be discretized using the uniformly stable Taylor-hood elements, described in [11, 21]. We will simulate the eruption that occurs at the interface between the viscous and porous domains with increased permeability. The Stokes equation will be used for describing the blood flow in the blood vessel, while the Darcy equation describes the flow in the surrounding tissue.

This thesis has been organized as follows. In chapter 2, the necessary anatomical and physiological background is provided, and an in-depth explanation of the medical problem. The mathematical model is presented in chapter 3. In

chapter 4, the finite element method and the implementation in FEniCS is introduced. The simulations and results are shown in chapter 5. In chapter 6, the results are summarized and discussed.

The automated software for solving partial differential equations (PDE), FEniCS [60], version 2019.1.0, is used for the numerical experiments. Gmsh [15] has been used as the finite element mesh generator software.

Chapter 2

Medical Background

In this chapter, a general description of the circulatory system in mammals and fish are introduced, and an approach to illustrate the different characteristics of blood flow. Furthermore, available experimental data provided by Vollestad [56, 57] and Resseguier [41, 42, 43, 44] are integrated into the thesis.

2.1 The Circulatory System

The circulatory system is a vast network of organs and blood vessels which provides oxygen and other nutrients to tissues in the body, and removes unnecessary waste outputs. In this section, the general understanding of the circulatory system in mammals and fish are explained individually.

2.1.1 Mammals

The heart is a very complex muscle that pumps the blood into three systems; the coronary, pulmonary, and systemic circuit (see fig. 2.1.1). Each of these systems performs an essential function for the circulatory system.

The coronary circulation is located in the heart, where it takes blood directly from the main artery surrounding the heart. Then it diverges into capillaries, which supply the heart muscles with oxygen, and the cardiac veins drain the blood once it has been deoxygenated. The next system in the circulation is the pulmonary circuit. The pulmonary circulations work only between the heart and lungs. The deoxygenated blood will travel to the lungs where it will *absorb* oxygen and release the carbon dioxide (CO_2), eventually bring the oxygenated blood back to the heart [34]. The systemic circuit utilizes the arteries as transports system for the blood to the organs and tissues. The arteries will split into arterioles, before it then divides into more capillaries. The capillaries will provide oxygen and nutrients to the tissues, while the waste and carbon dioxide

will be picked up by the venules. At this step, the circulatory system are completed, and the venues will bring the waste back to the pulmonary system (see fig. 2.1.1).

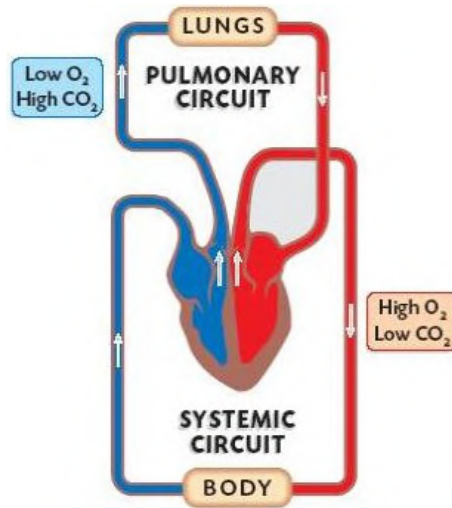


Figure 2.1.1: An illustration of the pulmonary and systemic circuits, with a view of the amount of oxygen and waste. Image from [54].

2.1.2 Zebrafish

The circulatory system in a fish is divided into gill, systemic and coronary circulation, shown in fig. 2.1.2.

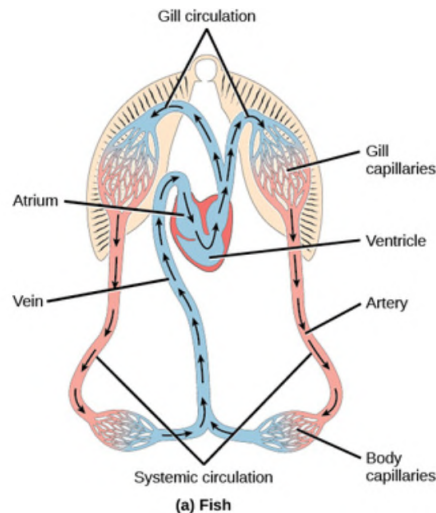


Figure 2.1.2: An illustration of the circulatory system in fish. Image from [[34], Chapter 20.2].

The heart consists of four parts: sinus venosus, atrium, ventricle, and bulbus arteriosus. The sinus venosus acts like a storage chamber for the oxygenated

blood. The sinus venosus contains the pacemaker cells responsible for initiating the contractions, such that the blood pumps into the atrium. Then the atrium generates weak contractions such that the blood is pushed into the ventricle. The ventricle transports the blood directed into the bulbus arteriosus, which is a small chamber that contains elastic components, but not many muscles. The atrium and ventricle have different attributes. The atrium will collect the blood that has returned from the body, while the ventricle will pump the blood to the gills. The ventricle is responsible for the generating of enough pressure, such that the blood can travel throughout the entire body. Bulbus arteriosus has few purposes, where it includes damping the pressure from the ventricle, such that there is no damage to the thin-walled gills under the circulations. The gills are the primary respiratory organs of fish. The gill consists of branched tissues richly supplied with blood vessels, especially near the gill surface. The gill also assists the progress of exchanging oxygen and carbon dioxide with the surrounding water. Then systemic circulation transports the oxygenated blood and provides oxygen and other nutrients to the cells. It also absorbs carbon dioxide and waste materials, which is carried to sinus venosus before it is transported into coronary circulation [19].

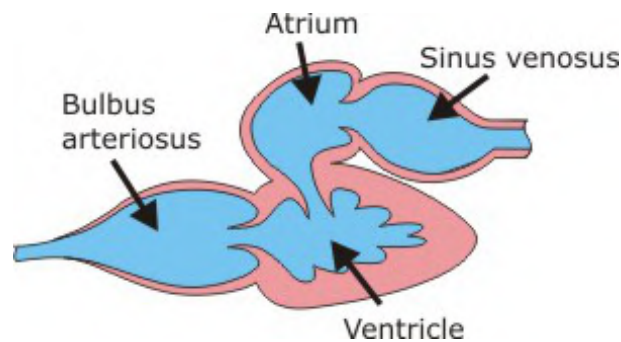


Figure 2.1.3: The different systems in a fish heart; bulbus arteriosus, atrium, ventricle, and sinus venosus. Image from [19].

2.1.3 Similarities Between Mammals and Fish

The general differences between mammals and fish are given as follows:

- The mammals have a double circulation, whereas fishes have a single circulation.
- Mammals have lungs, whereas fishes have gills.

Another difference that we have not focused on is the temperature. The fish are cold-blooded, and mammals are warm-blooded. The fish's overall body temperature dramatically varies due to environmental changes, while mammals always try to keep the body at the same temperature. The blood flow in our circulation will always be the same temperature, while the fish will change the temperature with the environment.

There is some similarity between the circulation system in fish and mammals, despite the gills and the lungs are very different when it comes to a physical standpoint, they still have some of the same functionality. They pull oxygen from the surrounding air or water and insert it into the bloodstream. Both circulate the deoxygenated blood to the gills or lungs, where they get dumped. The functionality of the heart is roughly the same, with the primary goal to pump blood throughout the entire body.

2.2 Flow in Blood Vessels

Now that we have a general view of how the circulation work, we can start with implementing some simple mathematical equation to describe the flow in the blood vessel.

The blood flow goes through three segments: arteries, capillaries, and veins. Arteries contribute blood to the entire body, losing very little pressure, and the diameter is almost uniform all over. After contributing the blood, capillaries will collect all that nutrients and wastes, which are exchanged between the blood and body tissues. The veins will dispose of the waste. From the introduction (see chapter 1), we explained that the blood vessels was one of the leading transit operator for cancer cells. The cancer cells travel through the arteries, and spread to random locations in the body. The interior surface of a blood vessel is formed by endothelial cells. Endothelial cells form a barrier that controls the flow of substances and fluid in and out of a tissue. The structure of a blood vessel is explained more thoroughly in the paper by Betts et al. [2].

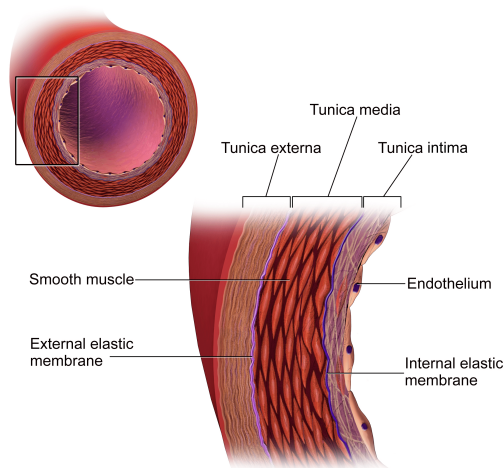


Figure 2.2.4: An illustration of the structure in an artery wall. Image from [17].

2.2.1 A Real Case of an Eruption

In this section, we will analyze the pictures provided by Resseguier [41, 42, 43, 44], where a healthy zebrafish have been injected with nanoparticles, which have

been used for illustrating tumor carrying particles.

The images that we will study in this section shows us how the cell structure is inside a zebrafish. The images are taken from the same zebrafish, where they illustrate different states of inflammation in an arbitrary area. Figures 2.2.7 and 2.2.8 are obtained from an area in proximity to the disease (granuloma). The cross-section in each image are chosen such that we get a view over three vessels in each case. Figure 2.2.5 shows cells and vessels of a healthy part of zebrafish, with a system of tissues/muscles around the vessels. An accurate description of the parts in the images is that the porous media are represented with tissues/muscles, while viscous media are represented with the flow in the vessel. The porous and viscous domain are described thoroughly in chapter 3. The images below show different steps of an eruption in healthy and infected zebrafish. Some of the injected nanoparticles will leak with the eruption, and spread by the junctions. Note that the eruptions also occur on a lesser amplitude. The spread of an eruption acts differently for each image, shown in figs. 2.2.5 to 2.2.8.

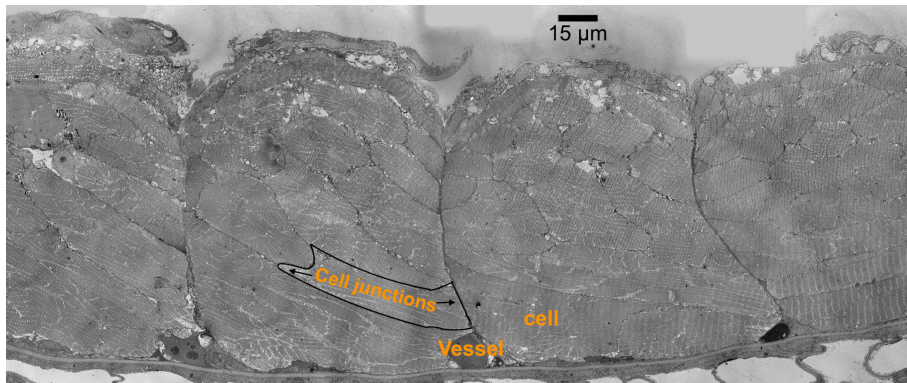


Figure 2.2.5: A description of the different parts in a healthy zebrafish. The cell junctions separate the cells, and the blood vessel are represented with circles. Image from [41].

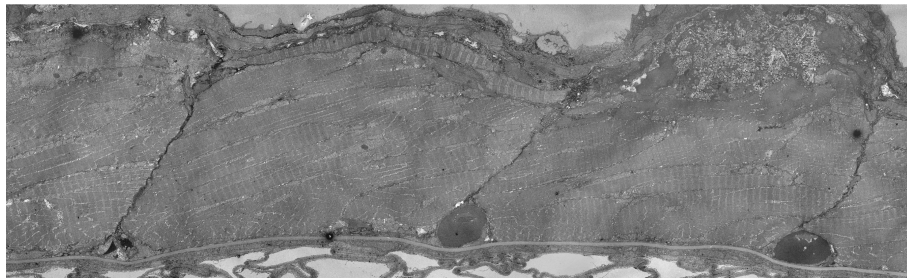


Figure 2.2.6: An illustration of the beginning of an eruption. The nanoparticles have started to burst out from the blood vessels, and the spreading has started from the left and right blood vessels. We see some signs of a granuloma on the top right corner. Image from [42].

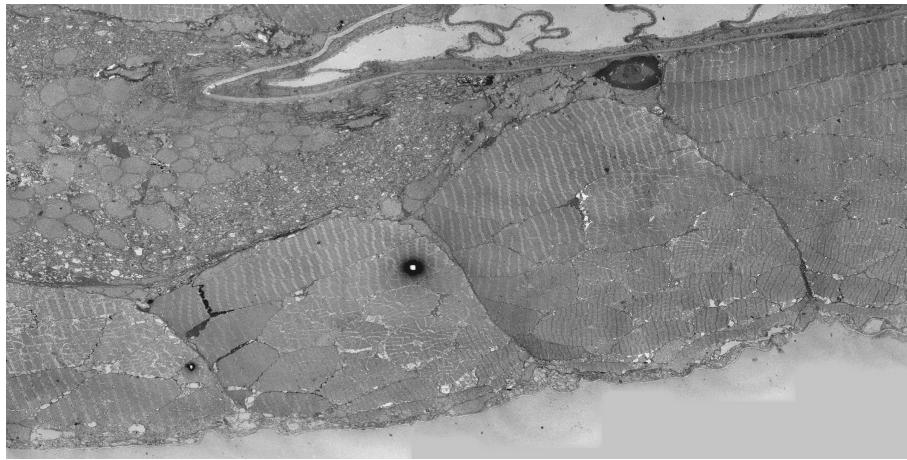


Figure 2.2.7: This image illustrates a developed eruption case on a much bigger scale. The eruption has spread, and numerous cells have been infected. There is still a visible blood vessel, while the other vessels have dissolved. The granuloma has taken over most of the cells, especially those located on the top left corner. Image from [43].



Figure 2.2.8: An illustration of fully dissolved blood vessels. The inflammation has reached most of the cells. Image from [44].

Figures 2.2.7 and 2.2.8 shows signs of granuloma, which is formed during the inflammation. Figures 2.2.5 to 2.2.8, provides information on how nanoparticles spread from a blood vessel into the tissues in an observed experiment. We will analyze another experiment where eruptions occur, section 2.4.2, which will be associated with the images shown in this section.

2.2.2 Viscosity

Viscosity is a valuable fluid property for analyzing the fluids behavior and fluid motion near solid boundaries. Viscosity is referred to as the quantity that describes a fluid's resistance. It will decide the energy that is required to make the fluid move. Liquids that flow very slowly have high viscosity, while liquids with

low viscosity flow much faster. Viscosity at a molecular level can be described as an interaction between the different molecules or friction within a fluid.

Blood viscosity is used to measure the thickness and stickiness of an individual's blood. The viscosity is usually expressed in centipoise (cP), which is the equivalent to 1 millipascal second (mPa s). Measuring the blood viscosity on a micro-scale level is a challenging task. Thus results from the paper by Lee et al. [27] will be introduced for the blood viscosity alternatively. In the article, the blood viscosity is generated with the use of a microfluidic viscometer driven by surface tension. The result from the paper are shown in fig. 2.2.9. The scalar viscosity used throughout the thesis is the one closest to the fertilization stage, thus two days post-fertilization (dpf).

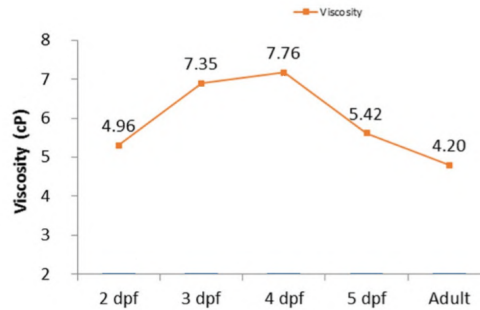


Figure 2.2.9: The development of blood viscosity from 2 dpf to an adult fish. Image from [27]

2.2.3 Poiseuille's Equation and Laminar Flow

Laminar flow is a reasonable flow conditions used for approximating the blood flow through some parts in the circulatory system. A Laminar flow can be described as a parabolic flow profile once it is fully developed. We assume large and straight blood vessels under steady condition, which result in an incompressible Newtonian fluid. The Poiseuille's equation will be introduced in this section for the calculation of the pressure difference for a laminar flow in a blood vessel.

The principles of conservation of mass, momentum, and energy need to be fulfilled for a blood flow, which states that whatever flows in must flow out. The conservation of mass and energy states that the total mass and energy in a system remain constant. The conservation of momentum states that momentum cannot be changed without any external force. The flows are drive by the gravitation and pressure gradient. It is essential to understand that the pressure gradient is the cause of blood move, and not the pressure itself. The forces that defy the flow are shear stresses due to the blood viscosity. The blood cells that are rolling and translation (tangential force) over the endothelial cells, will obtain shear stress at the boundary. The speed of the fluid at the endothelial cells is zero, relative to the boundary. The velocity increases from zero at the

wall to a maximum, in some height from the boundary (see fig. 2.2.10). Shear rate describes the rate of the neighboring layers of fluid moving concerning each other. Furthermore, this leads to a model of the velocity profile and result in the formulation of the velocity gradient. The pressure change is a valuable physical quantity that describes which direction and at what rate the pressure increases. The next step is to introduce a method to calculate the pressure gradient. Con-

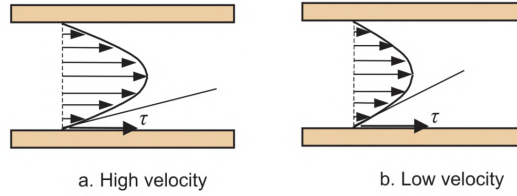


Figure 2.2.10: An Illustration of the velocity for different shear rates in a cylindrical tube. Image from [38].

Considering the blood vessel as a straight, cylindrical tube with a solid wall, then the shear rate is given as following [38]:

$$\gamma = \frac{du}{dr} \quad (2.1)$$

Where u represent the fluid velocity, and r is the radius of the cylindrical tube. The unit used for shear rate is $\frac{1}{s}$, where s stands for seconds. The equation applied for calculating the average shear stress is given by:

$$\tau = \frac{F}{A}$$

τ denotes the shear stress, F the applied force, and A denotes the cross-section area of the material, parallel to the applied force vector. The connection between shear stress and shear rate deepens on the fluid, which is the next step in the calculation.

The Ostwald de Waele equation [51] is introduced as a type of generalized Newtonian fluid, where the shear stress is presented as:

$$\tau = \kappa \cdot \gamma^n = \kappa \cdot \left(\frac{du}{dr}\right)^n \quad (2.2)$$

Where κ is the fluid internal properties, and n is the flow behavior index. Different values of n give different type of fluid:

$n < 1$	Non-Newtonian (Shear-thinning)
$n = 1$	Newtonian
$n > 1$	Non-Newtonian (Shear thickening)

Newton's viscosity laws are introduced to generate the shear stress for a stream-

ing Newtonian fluid:

$$\tau = \mu \frac{du}{dr}$$

The shear stress, τ is measured in Pascal (presented in eq. (2.1)), the fluid velocity is denoted as u , $\frac{du}{dr}$ is the velocity gradient in the horizontal direction of the flow (shear rate), and μ is the kinetic viscosity which is measured in pascal-seconds ($Pa \cdot s$).

In nature, a flow can be laminar, transitional, or turbulent. Turbulent flow defines a chaotic change in pressure and velocity, in both space and time. Laminar flow (or streamline flow) is the opposite, which occurs when fluids streams in parallel layers, with no interruption. This results in particles that flow in straight lines, parallel to the wall. The transitional flow is a mixture of laminar and turbulent flows. A way to calculate the fluid type, is to introduce Reynolds number:

$$Re = \frac{\rho v L}{\mu}$$

Where L is the length of the tube, ρ is the density, v is the velocity, and μ is the dynamic viscosity of the fluid. Different values of Reynolds number results in different types of fluid flow:

Re < 2300	Laminar flow
2300 < Re < 4000	Transition flow
Re > 4000	Turbulent flow (not necessarily)

The connection between pressure, flow, and resistance can be described as hemodynamic. Hemodynamics is based on a fundamental law of physics, namely Ohm's law:

$$I = \frac{\Delta V}{R}$$

Ohm's law is used to calculate the relationship between voltage differences (ΔV), current (I), and resistance (R). The Ohm's law are related to a fluid flow, where the voltage difference is the pressure difference, the current is the blood flow, and the resistance can be related to the resistance to flow, which is granted by the blood vessel and its interaction with the flowing blood [23]. The connections can be reviewed as following:

$$Q = \frac{\Delta p}{R} = \frac{P - P_0}{R} \quad (2.3)$$

$P - P_0$ is the pressure difference at the start to end, Q is the flow rate, and R is the resistance. Proposing that the electric and fluid resistance is proportional to the length of the system, and we have a non-slip condition at the wall. Thus a difference will occur in the electric and fluid. The fluid resistance is given by:

$$R = \frac{8\mu L}{\pi r^4} \quad (2.4)$$

μ denotes the dynamic viscosity, L the length of the pipe, and r is the radius of the pipe. With some rewriting and insertion of eq. (2.4) into eq. (2.3) we get the Poiseuille's equation:

$$\Delta p = \frac{8\mu L Q}{\pi r^4} \quad (2.5)$$

Now that we have calculated the Poiseuille's equation and accomplished a connection to the flow inside a blood vessel, we can start to calculate the pressure gradient for the circulatory system in a zebrafish. The viscosity measurements of a zebrafish, two days post-fertilization are introduced in the article by Chou et al. [4] (see fig. 2.2.9), where he concludes that the viscosity is $\mu = 4.96$ cP at $27^\circ C$. The volumetric flow rate Q can be described by:

$$Q = v \cdot A$$

where v denotes the flow velocity, and A is the cross-sectional vector area/surface, which can be described as πr^2 .

The flow velocity inside the blood vessel is studied and calculated in the paper by Vollestad [56], where PTV (particle tracking velocimetry) is introduced to follow nanoparticles through a zebrafish embryo. PTV is implemented for calculation of experiments with different sizes of nanoparticles in a zebrafish, with numerical methods, resulting in an overview of the velocity. In conclusion, the value $v = 300 \frac{\mu m}{s}$, which we will accept as a standard value for velocity inside the blood vessel in a zebrafish.

The last variables we need to verify before calculating the pressure change, is the radius of a blood vessel, since this value varies, a mean value will be used. The diameter of the blood vessels can be measured from the images provided by Resseguier [41, 42, 43, 44], shown in section 2.2. To get the right scaling, we will use ImageJ [49]. ImageJ is a software used for image processing and analyzing. The length of a fish at this stage is around 6 mm (obtained from Resseguier [41, 42, 43, 44]), and since the blood vessel circulates back to the heart, we will assume the length of the circulation is around 12 mm.

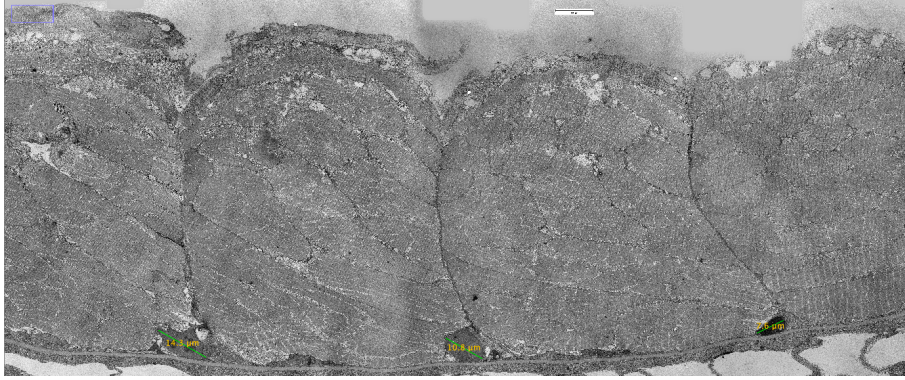


Figure 2.2.11: Visualsaiong of different diameter in some healthy blood vessels. The scaling bar is set to μm .

The next step is to implement the values into the Poiseuille's equation and calculate the pressure gradient:

$$\begin{aligned}\Delta p &= \frac{8\mu LQ}{\pi r^4} = \frac{8\mu Lv\pi r^2}{\pi r^2} = \frac{8\mu Lv}{r^2} \\ &= \frac{8 \cdot 4.96 \cdot 10^{-3} \frac{kg}{ms} \cdot 0.012m \cdot 300 \cdot 10^{-6} \frac{m}{s}}{(15 \cdot 10^{-6}m)^2} \\ \Delta p &= 634 \frac{kg}{ms^2} = 634 Pa\end{aligned}\tag{2.6}$$

With the blood vessel diameter set to $15 \mu m$, and the length of the circulation system in the zebrafish set to $0.012 \mu m$, results in a pressure gradient $634Pa$.

2.3 Permeability

Vascular permeability determines the capacity of a blood vessel wall, which allows for the flow of small molecules or even tumor cells in and out of the vessel. The blood cell wall (recognized as endothelial cells) have small gaps between the cells that are strictly regulated depending on the physiological state and type of the tissue. The endothelial barrier function is increased or decreased based on several mediators, which control the dynamically endothelial permeability. In healthy tissues, endothelial cells conserve basal vascular permeability at a low level, standard. When the infection starts, a messenger, inflammatory mediator, promotes an inflammatory response, which increases vascular permeability to induce plasma leakage [16, 33].

The permeability of tumor cell infected blood vessels includes a dynamic phenomenon characterized by vascular bursts followed by the brief, vigorous outward flow of fluid into the interstitial tumor space. The burst of fluid is often named eruptions, which are used throughout the thesis. The speculation behind

an eruption of fluid that burst into the interstitial space is the rapid increase of blood flow and the increase of permeability. A description of the different cases with increasing permeability is in Figure 2.3.12.

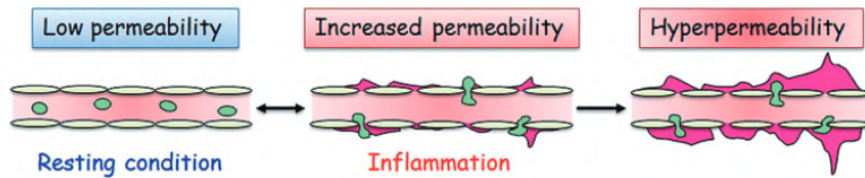


Figure 2.3.12: An illustration of endothelial permeability. The junctions are tight when it comes to low permeability, but increasing permeability will cause plasma leakage and, in the worst-case scenario, go over to hyperpermeability, where the separation between the junctions will lead to chronic inflammation, asthma, cancer, and other diseases. Image from [45]

2.4 Eruptions in Blood Vessels

In this section, we will look at some eruptions obtained by Vollestad [57]. The results in the report will have a big role in the thesis. They will be used for comparison with the numerical simulations accomplished in chapter 5.

2.4.1 Eruption

The vascular endothelial cells in the interior surface of blood vessels, behaves like a barrier between the viscous and porous domain. The viscous and porous domains are described in chapter 3. Abnormal vascular walls with leaky cell-to-cell junctions results in enhanced permeability. This will form temporary openings and closings between the endothelial cells. Furthermore, in healthy blood vessels, the endothelial cells receive signals such that entrances are formed in the vasculature as nutrients and oxygen are transported into the stromal space (porous domain) [48]. In a tumor infected blood vessel, *fake* signals will be sent and lead to leakage of tumor cells in the stromal space. The post-capillary venules reabsorb most of the fluid that leaks from the blood vessel. While in an unhealthy zebrafish, a portion of the fluid will filter through and leak into the tissues. This describes how tumors or other diseases spread in the body. An increase of fluid will occur when the zebrafish has received tumor cells. Thus more interstitial fluid will erupt into the healthy tissues. A stationary model of the development of eruption is described in figs. 2.2.5 to 2.2.8

A better representation of the eruption with nanoparticles can be seen in Vollestad [57]. The images used for calculations are received from the scanning light-sheet video by *Julien Resseguier* (described in [57]). Where nanoparticle *burst* following the extrusion of a white blood cell from a nano-fish blood vein has been observed. The images show a blood vessel in a healthy zebrafish that is infected by cancer cells, before and after an eruption occurs. Figure 2.4.13 and 2.4.14 demonstrate two different cases, where an arbitrary eruption emerges in the

blood vessel. The goal is to introduce a way to calculate the pressure difference for an eruption.

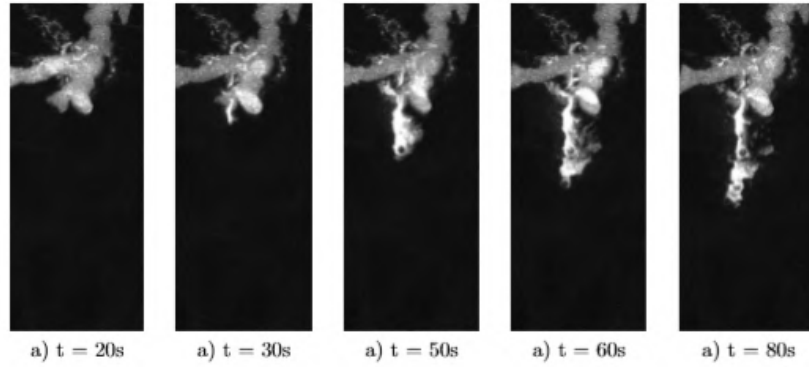


Figure 2.4.13: Time development of a laminar burst in a blood vessel. Image from [57].

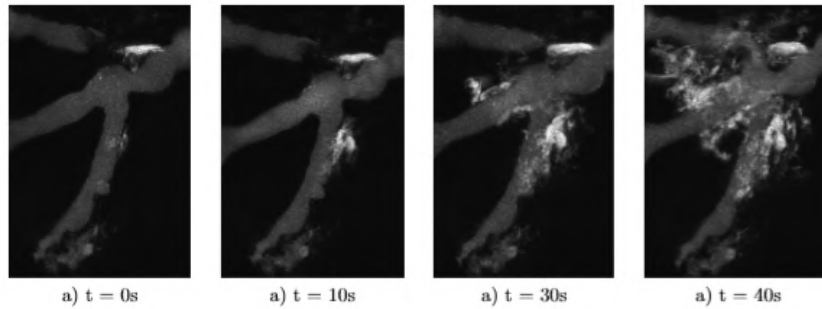


Figure 2.4.14: Time development of a radial burst in a blood vessel. Image from [57]

2.4.2 Laminar Eruption

Figure 2.4.13 illustrates one arbitrary eruptions. Image *a)* shows a normal blood vessel, where nanoparticles are shown as white. The tissues are shown as black, surrounding the blood vessel. When the time surpasses 20 seconds, the eruption starts to develop, which indicates that the permeability probably has increased, and the event where contraction between endothelial cells starts to take action, which we explained in section 2.3. The eruption will expand into the porous domain gradually. When the time hits around 80 seconds, the eruption starts to stabilize, and the junction starts to close up [33]. We assume that the eruptions have leaked out into a cell-to-cell junction between the tissues, which results in the straight expansion. Those cell-to-cell junctions are visible in the images shown in section 2.2.1.

Three-dimensional conversion of the first and last time-step of fig. 2.4.13:

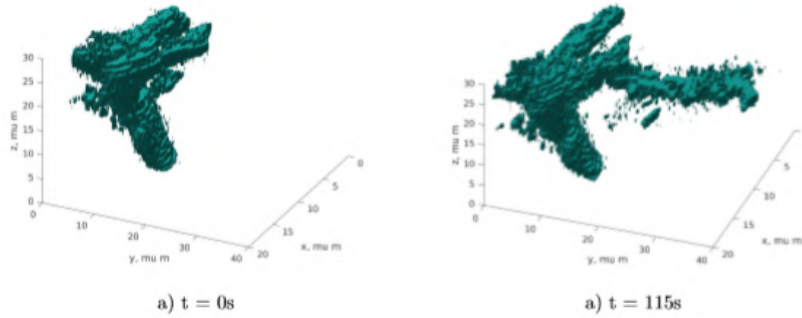


Figure 2.4.15: A 3D conversion of the laminar case. Image from [57]

Considering the eruptions expand out in only y-direction, we can model the fully developed eruption, in this case, a incompressible, Newtonian fluid with a laminar flow. This eruption can be represented weakly with a Poiseuille's flow. Importing the images into ImageJ, will let us measure the length and diameter of the eruption shown in fig. 2.4.15. The diameter development through the eruption is shown in fig. 2.4.16. The length of the eruption is $30 \mu m$, which is the final length for the eruption.

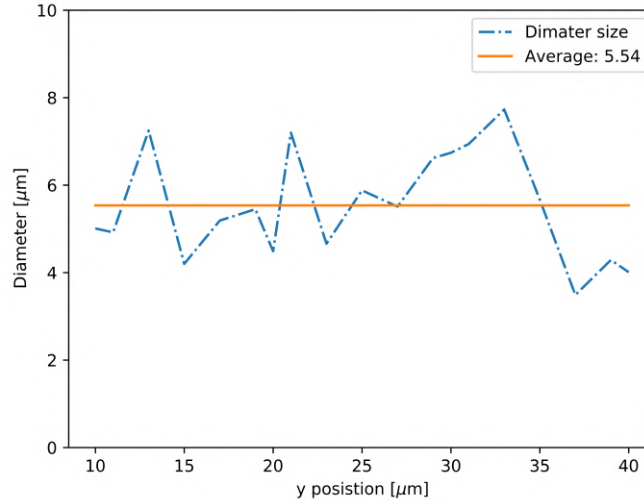


Figure 2.4.16: Diameter development through the stabilized eruption.

This provides us with the mean diameter equal to $5.4 \mu m$, and the mean velocity equal to $0.22 \frac{\mu}{s}$. The mean velocity is calculated with the same method performed for the diameter. These estimations are calculated from values obtained from the report by Vollestad [57]. We assume the eruption is tube-shaped when it is fully developed. Hence the Poiseuille's equation (eq. (2.5)) can be

applied to calculate the pressure difference.

$$\begin{aligned}\Delta p &= \frac{8\mu LQ}{\pi r^4} = \frac{8\mu Lv\pi r^2}{\pi r^2} = \frac{8\mu Lv}{r^2} \\ &= \frac{8 \cdot 4.96 \cdot 10^{-3} \frac{mg}{\mu ms} \cdot 30\mu m \cdot 0.22 \frac{\mu m}{s}}{(2.8\mu m)^2} \\ \Delta p &= 0.03340 \frac{mg}{\mu ms^2} = 0.03340 \text{ Pa}\end{aligned}\tag{2.7}$$

The pressure difference in the eruption is around $0.03340Pa$. This value will be used for comparison with the numerical experiments in chapter 5.

2.4.3 Radial Eruption

Figure 2.4.14 shows another potential eruption development that could occur. We refer to this as a radial eruption. A few more calculations need to be taken into consideration before we can calculate the pressure gradient. The 3D conversion of the first and last timestamp for this case is given by:

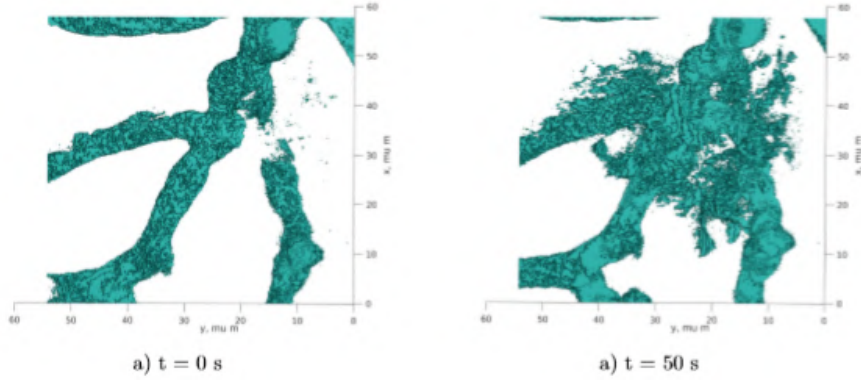


Figure 2.4.17: A 3D conversion of the first and last step of a radial eruption. Image from [57]

Womersley Flow

The goal is to generate a realistic inflow boundary condition which can apply for the radial eruption, when the pressure gradient is unknown. We assume a flow with periodic variations, known as pulsatile flow, or as Womersley flow [39]. A pulsatile flow implies that the pressure gradient varies periodically in time; this can be used for mimicking a blood flow in the arteries. We simplify the problem with the following assumption:

- Homogeneous, incompressible and Newtonian fluid.
- Circular and rigid wall tube.
- No gravitation force on the fluid.

- Axisymmetry at the center and no-slip on the wall, boundary condition.

Those assumptions will give us the simplified Navier-Stokes and continuity equation:

$$\begin{aligned} \mu \left(\frac{\partial^2 w}{\partial r^2} + \frac{1}{r} \frac{\partial w}{\partial r} \right) - \frac{\partial p}{\partial x} &= \rho \frac{\partial w}{\partial t} \\ \frac{\partial w}{\partial x} &= 0 \end{aligned} \quad (2.8)$$

w represents the longitudinal velocity of the fluid at a distance $0 \leq r \leq a$, a represents the most distant radius from the center. We assume that the pressure gradient is periodic in time and of the form:

$$\frac{\partial p}{\partial x} = -Ae^{in\omega t}$$

with $i = \sqrt{-1}$, n is a positive integer, t as time. A and ω are positive real numbers. We can decompose the velocity similarly, with separation of variables:

$$w(r, t) = u(r)e^{in\omega t} \quad (2.9)$$

Inserting this into (2.8) gives us the Navier-Stokes equation for each harmonic:

$$\begin{aligned} \left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} \right) e^{in\omega t} + \frac{A}{\nu} e^{in\omega t} &= \frac{in\omega}{\nu} u e^{in\omega t} \\ \frac{\partial^2 u}{\partial r^2} + \frac{1}{r} \frac{\partial u}{\partial r} + \frac{A}{\nu} &= \frac{in\omega}{\nu} u \end{aligned} \quad (2.10)$$

where $\nu = \frac{\mu}{\rho}$ represents the kinematic viscosity. The boundary conditions are satisfied, with the axisymmetric boundary condition $\frac{\partial u}{\partial r}|_{r=0} = 0$, results in:

$$u(r) = \frac{A}{in\omega} \left[1 - \frac{J_0 \left(r \sqrt{\frac{i^3 n \omega}{\nu}} \right)}{J_0 \left(a \sqrt{\frac{i^3 n \omega}{\nu}} \right)} \right]$$

J_0 represents a first-order Bessel function. This will lead to the Womersley equation with the Womersley number $\alpha = a \sqrt{\frac{n\omega}{\nu}}$:

$$w(r, t) = \frac{A}{in\omega} \left[1 - \frac{J_0(\beta_n y)}{J_0(\beta_n)} \right] \quad (2.11)$$

$\beta_n = \alpha_n \sqrt{i^3}$ and $y = \frac{r}{a}$. We need to apply some modifications to (2.11) since it requires that we know the pressure gradient as a function of time. A periodic function Q can be used as an approximation, with N complex Fourier models $\{Q_n\}$, this can only be done if we consider a case where the flow rate $Q(t)$ is a periodic function of time with period T :

$$\omega = \frac{2\pi}{T}, \quad Q(t) = \sum_{n=0}^N Q_n e^{in\omega t} \quad (2.12)$$

Now we will assume a solution on the form:

$$v(t, y) = \sum_{n=0}^N C_n Q_n \left(1 - \frac{J_0(\beta_n y)}{J_0(\beta_n)} \right) e^{in\omega t}$$

C_n needs to be chosen such that the flow rate function is reproduced: $\int_{\Gamma} v ds = Q(t)$. Hence we have the identity:

$$\sum_{n=0}^N \left(C_n Q_n 2\pi a^2 e^{in\omega t} \int_0^1 y \left(1 - \frac{J_0(\beta_n y)}{J_0(\beta_n)} \right) dr d\theta \right) = Q(t)$$

Since we have a complex exponential that are orthogonal in time:

$$\begin{aligned} C_n 2\pi a^2 \int_0^1 y dy - C_n 2\pi a^2 \int_0^1 y \left(\frac{J_0(\beta_n y)}{J_0(\beta_n)} \right) dy &= 1 \\ C_n \pi a^2 - C_n 2\pi a^2 \int_0^1 y \left(\frac{J_0(\beta_n y)}{J_0(\beta_n)} \right) dy &= 1 \\ \frac{1}{C_n} &= \pi a^2 \left(1 - \frac{2}{J_0(\beta_n)} \int_0^1 y \left(\frac{J_0(\beta_n y)}{J_0(\beta_n)} \right) dy \right) \end{aligned} \quad (2.13)$$

$$\frac{1}{C_n} = \pi a^2 \left(1 - \frac{2J_1(\beta_n)}{\beta_n J_0(\beta_n)} \right) \quad (2.14)$$

The transaction between (2.13) and (2.14) is done by the Bessel property:

$$\frac{d}{dx} [x J_1(ax)] = ax J_0(ax)$$

hence we have the final solution, from the given flow rate function $Q(t)$:

$$V_w(t, y) = \frac{2Q_0}{\pi a^2} (1 - y^2) + \sum_{n=0}^N \frac{Q_n}{\pi a^2} \frac{1 - \frac{J_0(\beta_n y)}{J_0(\beta_n)}}{1 - \frac{2J_1(\beta_n)}{\beta_n J_0(\beta_n)}} e^{in\omega t} \quad (2.15)$$

(2.15) is the best realistic inflow boundary condition we can apply for the radial eruption, when the only known data is the volumetric flow rate. The volumetric flow rate for Figure 2.4.17 can easily be calculated using the results from [57]. (2.15) is the best realistic inflow boundary condition we can apply for the radial eruption, when the only known data is the volumetric flow rate.

Chapter 3

Mathematical Model

In this chapter, we will present the mathematical model applied for coupled Darcy-Stokes flow. Considering flow in a domain $\Omega \in \mathbb{R}^2$, which is separated into two non-overlapping subdomain. Porous domain denoted as Ω_d , and viscous domain denoted as Ω_s , with the interface Γ .

3.1 Viscous Flow

The blood flows inside a vessel are initially represented as non-Newtonian fluid, however the mathematical aspect for this very complicated. However, In this thesis, the blood flow is assumed to be a Newtonian fluid, which makes the mathematical model much simpler. This was also introduced in section 2.2.3. However, this is still valid, since the blood flow inside a large vessel acts like a Newtonian fluid. The mathematical equation describing the flow in the viscous domain is derived from the Navier-Stokes equation.

3.1.1 The Navier-Stokes Equation

Navier-Stokes equations are used to describe the motion of an incompressible Newtonian fluid. In a Newtonian fluid, the connection between shear stress and shear rate is proportional. Hence, the viscosity is constant. Incompressible fluid states that the fluid density (ρ) is constant. For the full derivations of the Navier-Stokes equation, see, e.g. [[59] chapter 2].

$$\rho \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot \sigma(\mathbf{u}, p) + \mathbf{f} \quad (3.1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (3.2)$$

ρ represents the density of the fluid, μ denotes the viscosity, \mathbf{f} the body forces. The variables that we are interested in are the velocity field, and the pressure

field denoted as \mathbf{u} and p , respectively. $\sigma(\mathbf{u}, p)$ is the Cauchy stress tensor, which is presented as:

$$\sigma(\mathbf{u}, p) = 2\mu\epsilon(\mathbf{u}) - p\mathbb{I} \quad (3.3)$$

$\epsilon(\cdot)$ represents the symmetric strain rate tensor:

$$\epsilon(\cdot) = \frac{1}{2}(\nabla(\cdot) + (\nabla(\cdot))^T) \quad (3.4)$$

Equation (3.1) is referred to as the *equation of motion*. Equation (3.2) is derived from the conservation of mass within the system and is referred to as the *equation of continuity*.

3.1.2 Stokes Equation

Stokes flow is described as a fluid with low Reynolds numbers ($\text{Re} \ll 1$), where the convective term $(\mathbf{u} \cdot \nabla)\mathbf{u}$, are assumed to be neglected compared to the viscous forces. Since we are assuming a stationary problem, we get that $\frac{\partial \mathbf{u}}{\partial t} = 0$. As a result, we are left with a simplification of the Navier-Stokes equation, i.e., the linear stationary Stokes equations:

$$-2\mu\nabla \cdot \epsilon(\mathbf{u}) + \nabla p = \mathbf{f} \quad \text{in } \Omega_s \quad (3.5)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega_s \quad (3.6)$$

$$\mathbf{u} = 0 \quad \text{on } \partial\Omega_s^D \quad (3.7)$$

$$(2\mu\epsilon(\mathbf{u}) + p\mathbf{I})\mathbf{n} = 0 \quad \text{on } \partial\Omega_s^N \quad (3.8)$$

Denoting viscosity as μ , strain-rate tensor as ϵ , the body force as \mathbf{f} , velocity field as \mathbf{u} and the pressure as p . \mathbf{n} is the unit outward normal vector on the boundary. The Dirichlet condition defines the value on the surface, which is set in eq. (3.7). The Neumann condition is set in eq. (3.8), specifying the normal derivatives of the function at the surface. Equation (3.7) is set to zero, since we are assuming no slip at boundaries. \mathbf{I} represent the identity matrix:

$$\mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

3.2 Porous Flow

Porous media are solid materials that are composed of pore structures with an interconnected void, which allows fluid to pass through. A porous medium is connected with an extensive network of pores with the complex flow, and this makes it impossible to achieve a detailed description of the flow field. The tissue surrounding a blood vessel can be classified as a porous media. An example of cells with junction in zebrafish are shown in section 2.2.1. The fluid flow through the porous media pursues the same principals as a free viscous flow, and the guiding equation can be obtained from the Navier–Stokes equation via homogenization. Darcy’s equation is generally accepted as the macroscopic equation of motion for Newtonian fluids in porous media at small Reynolds numbers (Stokes flow). For the full derivations of Darcy’s law, see [36].

3.2.1 Darcy's Equation

Darcy's law representing the incompressible porous media flow:

$$\mathbf{q} = -\frac{\mathbf{K}}{\mu}\nabla p \quad (3.9)$$

The equation presents a proportional relationship between the volumetric flux \mathbf{q} , referred to as the Darcy flux, and ∇p denotes the pressure difference. μ denotes the viscosity of the fluid, and \mathbf{K} denotes the permeability tensor. The permeability tensor is used to measuring the ability for a porous material to permit fluid to pass through it. For simplicity, the porous medium is assumed to be homogeneous and isotropic, which allows the permeability to be represented by a scalar K . In reality, most porous materials are both heterogeneous and anisotropic. \mathbf{q} denotes the Darcy flux (or Darcy velocity). The relation between the velocity and flux is defined by the porosity (ϕ):

$$\mathbf{u} = \frac{\mathbf{q}}{\phi} \quad (3.10)$$

The motion of an incompressible fluid through a saturated porous medium is described as a combination of Darcy's law (3.9), and the equation of continuity (3.2). This results in the Darcy's equation:

$$\frac{\mu}{K}\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega_d \quad (3.11)$$

$$\nabla \cdot \mathbf{u} = \mathbf{f}_2 \quad \text{in } \Omega_d \quad (3.12)$$

$$p = 0 \quad \text{on } \partial\Omega_d^D \quad (3.13)$$

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad \text{on } \partial\Omega_d^N \quad (3.14)$$

The source term \mathbf{f} has been added to Darcy's equation, such that we can take body force into account, \mathbf{n} denotes the unit outward normal vector at the boundary, which has non-overlapping decomposition. The Dirichlet condition is set in (3.13) and Neumann condition is set in (3.14), they have be set for the exterior boundaries. We assume that the source term \mathbf{f}_2 satisfies the condition specified in eq. (3.15), hence \mathbf{f}_2 are set to zero eq. (3.12).

$$\int_{\Omega} \mathbf{f}_2 \, d\Omega = 0 \quad (3.15)$$

This makes physical sense due to $\mathbf{u} \cdot \mathbf{n} = 0$ (no flow), and also satisfies the interface condition shown in eq. (3.20).

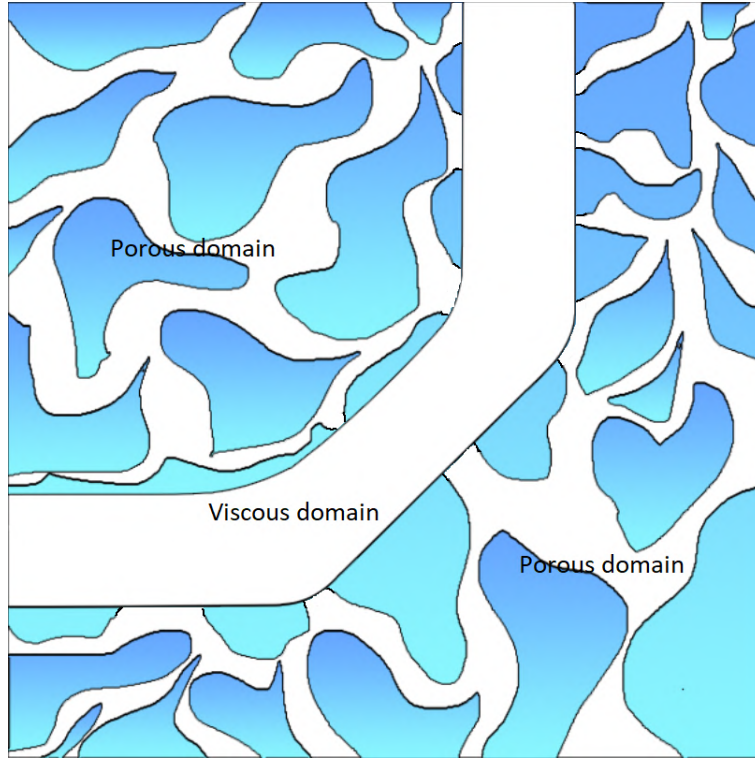


Figure 3.2.1: An illustration of the split domain with viscous and porous domain.

3.3 Coupled Darcy–Stokes Flow

The coupled Darcy–Stokes flow consists of the viscous and porous domain $\Omega = \Omega_d \cup \Omega_s$, where the porous region Ω_d is dependent by Darcy’s equation, and the open viscous region Ω_s is dependent by the linear stationary Stokes equations. The coupled Darcy–Stokes problem which is to find $(\mathbf{u}, p) \in \mathbf{V} \times Q$ such that:

$$-2\mu \nabla \cdot \epsilon(\mathbf{u}) + \nabla p = \mathbf{f} \quad \text{in } \Omega_s \quad (3.16)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega_s \quad (3.17)$$

$$\frac{\mu}{K} \mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega_d \quad (3.18)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega_d \quad (3.19)$$

denoting viscosity by μ , strain-rate tensor by ϵ , the body force by \mathbf{f} and the scalar permeability by K . The solution space for the velocity \mathbf{u} is \mathbf{V} , and the solution space for the pressure p is Q . The domains are coupled at the interface $\Gamma = \partial\Omega_d \cap \partial\Omega_s$. Following boundary conditions are stated at the interface:

$$\mathbf{u}_s \cdot \mathbf{n}_s = \mathbf{u}_d \cdot \mathbf{n}_d \quad \text{on } \Gamma \quad (3.20)$$

$$2\mu \mathbf{n}_s \cdot \epsilon(\mathbf{u}_s) \cdot \mathbf{n}_s = p_s - p_d \quad \text{on } \Gamma \quad (3.21)$$

$$2\mathbf{n}_s \cdot \epsilon(\mathbf{u}_s) \cdot \boldsymbol{\tau} = \alpha K^{-\frac{1}{2}} \mathbf{u}_s \cdot \boldsymbol{\tau} \quad \text{on } \Gamma \quad (3.22)$$

Equation (3.20) and (3.21) represent the mass conservation and the continuity of the normal stress, respectively. Equation (3.22) represent the Beavers-Joseph-Saffmann condition [20]. τ denotes the unit tangential vector at Γ , \mathbf{n}_s and \mathbf{n}_d is the unit normal vector exterior to Ω_s and Ω_d , α is a dimensionless coefficient determined by the structure near the interface. The \mathbf{u}_s and p_s represent the velocity and pressure in the open region Ω_s while \mathbf{u}_d and p_d represent the velocity and pressure in the porous region Ω_d .

In fig. 3.3.2, the two domains are illustrated with an overview of the interior boundary conditions and interface conditions:

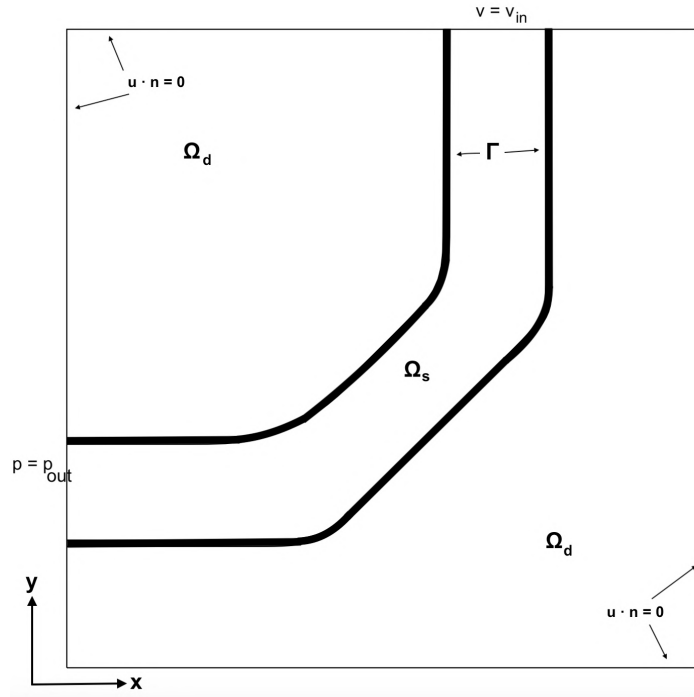


Figure 3.3.2: The domain is split into the viscous Ω_s and porous Ω_d subdomains, by the interface Γ .

The domain shown in fig. 3.3.2, consists of a porous and viscous subdomain represented with Ω_d and Ω_s , with the coupled interface conditions Γ from eqs. (3.20) to (3.22). The inlet and outlet Dirichlet conditions are represented as v_{in} and p_{out} .

The geometry of the domain is generated to imitate the images shown in figs. 2.2.5 to 2.2.8. Note that the images are shown from a different angle. The red square in fig. 3.3.3, symbolizes the boundary of the simulations done later in this thesis (in chapter 5).



Figure 3.3.3: Illustration of a region in a zebrafish, from section 2.2.1. We will focus on the area inside the red square, which is simplified, as shown in fig. 3.3.2.

Chapter 4

Numerical Methods

Solving complex elliptic and parabolic partial differential equations (PDEs) analytical is a challenging task, for example, the coupled Darcy-Stokes equation. Thus, numerical approximations are introduced. The numerical method used in this paper is the finite element method (FEM), which is one of the main tools for solving numerical problems [29]. The programming language used to solve the FEM is FEniCS [60], which is a software package designed to solve PDEs with high-level Python and C++. The theories presented in this chapter are collected from the work by Logg et al. [30], Ciarlet [6], and Elman et al. [12].

4.1 The Finite Element Method

A finite element method is a numerical technique introduced to form an approximate solution for partial differential equation problems. FEM can handle complicated geometries and boundaries with relative ease, which makes it a convenient method for flow simulations. Some advantages in the FEM:

- Modeling a complex physical deformation analytical can be ineffective, by hand. However, a high order of accuracy can be concluded when introducing the finite element method.
- The modeling of complex structure becomes easy in FEM. Thus, we can easily observe how critical factors affects the model.
- Useful for time-dependent simulations, such that the deformation in different areas can be observed.
- Boundary continuous are easily integrated in FEM.

However, there are a few disadvantages. Some examples, a large amount of data is required, high computational time usage, and only an approximate solution is obtained.

There are four steps in solving a PDE with FEM.

- Initialize the strong form of our problem.
- Define the weak form of the PDE.
- Discretize the domain.
- Choose the finite-dimensional function space.

The Poisson equation will be used as an example in the first part of this chapter to get an overview of the finite element method. The Poisson's equation is the simplest and the most well-known elliptic partial differential equation. Considering $\Omega \in \mathbb{R}^n$ with the boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$. The subscripts D and N represents the Dirichlet and Neuman boundary, respectively.

$$-\Delta \mathbf{u} = -\nabla^2 \mathbf{u} = f \quad \text{in } \Omega \in \mathbb{R}^2 \quad (4.1)$$

$$\mathbf{u} = u_0 \quad \text{on } \Gamma_D \quad (4.2)$$

$$\nabla \mathbf{u} \cdot \mathbf{n} = g \quad \text{on } \Gamma_N \quad (4.3)$$

f and g are given functions, \mathbf{u} is some unknown field, \mathbf{n} is the outward unit normal vector on the boundary, and ∇^2 is the Laplace operator. The Dirichlet boundary (4.3), defines a guiding value for the unknown field. While the Neumann boundary condition (4.2), defines a guiding value for the normal derivative. Equation (4.1) is referred to as the strong form of Poisson's equation.

4.1.1 Sobolev Spaces

One of the fundamental tools used for analyzing partial differential equations and the finite element method is Sobolev space. For more details of the weak derivative and general concept of Sobolev spaces, are shown in the textbook by Evans [13].

Definition 4.1.1. (L^p -spaces [[13], Appendix D]) Let Ω be an open subset of \mathbb{R}^n . For $1 \leq p \leq \infty$, $L^p(\Omega)$ is defined as:

$$L^p(\Omega) = \{\mathbf{u} \in \Omega \rightarrow \mathbb{R} : (\int_{\Omega} |\mathbf{u}|^p dx)^{\frac{1}{p}} < \infty\} \quad (4.4)$$

with the corresponding norm $\|\mathbf{u}\|_{L^p(\Omega)} = (\int_{\Omega} |\mathbf{u}|^p dx)^{\frac{1}{p}}$

Definition 4.1.2. Sobolev space [[13], Section 5.2]) Let Ω be an open subset of \mathbb{R}^n . For a non-negative integer m where $\alpha = (\alpha_1, \dots, \alpha_n)$ denotes a multi-index,

the Sobolev space $H^m(\Omega)$:

$$H^m(\Omega) = \mathbf{u} \in \Omega \rightarrow \mathbb{R} : \left(\sum_{|\alpha| \leq m} \int_{\Omega} |D^{\alpha} \mathbf{u}|^2 dx \right)^{\frac{1}{2}} < \infty \quad (4.5)$$

with the corresponding norm $\|\mathbf{u}\|_{H^m(\Omega)} = \left(\sum_{|\alpha| \leq m} \int_{\Omega} |D^{\alpha} \mathbf{u}|^2 dx \right)^{\frac{1}{2}}$

$D^{\alpha} \mathbf{u}$ denotes the α – derivatives of \mathbf{u} . Using functions from these spaces, clarifies that the integrals involved in the variational form are bounded.

4.1.2 Weak Formulation

The solution \mathbf{u} is ideal only if it has continuous second derivatives in Ω , which states that \mathbf{u} is in $\mathbb{C}^2(\Omega)$ and continuous up to the boundary. In the case where we have non-smooth boundaries or discontinuous source function, the function \mathbf{u} is not sufficiently smooth to hold the requirements of the strong solution. The weak form is then introduced to work with less restrictive problems. A test function $\mathbf{v} \in \hat{\mathbf{V}}$ is introduced to derive the weak formulation of the problem.

Multiplying the Poisson equation (4.1) with the test function and integrating the product of the strong form, results in the weak form:

$$- \int_{\Omega} \nabla^2 \mathbf{u} \mathbf{v} \, dx = \int_{\Omega} f \mathbf{v} \, dx \quad (4.6)$$

In this case, integration by part and the divergence theorem is essential for eq. (4.6). Introducing $\mathbf{v} = 0$ on the Dirichlet boundary Γ_D .

$$- \int_{\Omega} \nabla^2 \mathbf{u} \mathbf{v} \, dx = \int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx + \int_{\Omega} \nabla \cdot \mathbf{v} \nabla \mathbf{u} \, dx \quad (4.7)$$

$$= \int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx + \int_{\partial\Omega} \mathbf{v} \frac{\partial \mathbf{u}}{\partial n} \, ds \quad (4.8)$$

Applying the boundary condition for Neumann in eq. (4.1), we will result in: find $\mathbf{u} \in \mathbf{V}$ such that

$$\int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx = \int_{\Omega} f \mathbf{v} \, dx - \int_{\Gamma_N} g \mathbf{v} \, ds \quad \forall \mathbf{v} \in \hat{\mathbf{V}} \quad (4.9)$$

Equation (4.9) is referred to as the weak formulation, also known as the variational formulation for Poisson's equation. In generic form the Poisson equation can be written as follows:

$$a(\mathbf{u}, \mathbf{v}) = L(\mathbf{v}) \quad (4.10)$$

A common choice of the test and trial space $\hat{\mathbf{V}}$ and \mathbf{V} are defined by:

$$\mathbf{V} := \{ \mathbf{u} \in H^1(\Omega) : \mathbf{u} = u_0 \text{ on } \partial\Omega_D \} \quad (4.11)$$

$$\hat{\mathbf{V}} := \{ \mathbf{v} \in H^1(\Omega) : \mathbf{v} = 0 \text{ on } \partial\Omega_D \} \quad (4.12)$$

The test space $\hat{\mathbf{V}}$ (4.11) is defined similar to the trial space \mathbf{V} (4.12), but with a substituted Dirichlet condition.

4.1.3 Discretization

In order to solve the problem numerically, we need to discretize the variational problem by seeking an approximation, $\mathbf{u}_h \approx \mathbf{u}$, restricted to a finite-dimensional subspace $\mathbf{V}_h \subset \mathbf{V}$. The test function \mathbf{v} is restricted to $\hat{\mathbf{V}}_h \subset \hat{\mathbf{V}}$. This results in the discrete variational problem: find $\mathbf{u}_h \in \mathbf{V}_h$ such that

$$\int_{\Omega} \nabla \mathbf{u}_h \cdot \nabla \mathbf{v} \, dx = \int_{\Omega} f \mathbf{v} \, dx - \int_{\Gamma_N} g \mathbf{v} \, ds \quad \forall \mathbf{v} \in \hat{\mathbf{V}}_h \subset \hat{\mathbf{V}} \quad (4.13)$$

In order to solve the discrete variational problem, we need to introduce the basis functions, $\{\phi_j\}_{j=1}^N$ and $\{\hat{\phi}_i\}_{i=1}^N$, with an $N = \dim(\mathbf{V}_h)$, which denotes the degree of freedom. The basis functions can be used to span over the subspace:

$$\begin{aligned} \mathbf{V}_h &= \text{span}\{\phi_1(\mathbf{x}), \dots, \phi_N(\mathbf{x})\} \\ \hat{\mathbf{V}}_h &= \text{span}\{\hat{\phi}_1(\mathbf{x}), \dots, \hat{\phi}_N(\mathbf{x})\} \end{aligned}$$

The approximation u_h can be written as:

$$u_h(\mathbf{x}) = \sum_{j=1}^N U_j \phi_j(\mathbf{x}) \quad (4.14)$$

where U_j are coefficients, defining the degrees of freedom. However, requiring that (4.13) applies for all $\mathbf{v} \in \hat{\mathbf{V}}$ is equivalent to expecting that it apply for all basis functions.

$$\begin{aligned} \int_{\Omega} \nabla \left(\sum_{j=1}^N U_j \phi_j \right) \cdot \nabla \hat{\phi}_i \, dx &= \int_{\Omega} f \hat{\phi}_i \, dx - \int_{\Gamma_N} g \hat{\phi}_i \, ds, \quad i = 1, 2, \dots, N \\ \sum_{j=1}^N U_j \int_{\Omega} \nabla \phi_j \cdot \nabla \hat{\phi}_i \, dx &= \int_{\Omega} f \hat{\phi}_i \, dx - \int_{\Gamma_N} g \hat{\phi}_i \, ds, \quad i = 1, 2, \dots, N \end{aligned} \quad (4.15)$$

\mathbf{u}_h has been replaced by eq. (4.14). We assume a linear PDE such that the generic form $a(\mathbf{u}, \mathbf{v})$ is linear in \mathbf{u} , for simplicity. Hence the equation can be written in bilinear form by:

$$\sum_{j=1}^N a(\phi_j, \hat{\phi}_i) U_j = L(\hat{\phi}_i), \quad i = 1, 2, \dots, N \quad (4.16)$$

eq. (4.16) is identified as a linear algebraic system of eq. (4.15). Solving this is equivalent to solving the matrix equation:

$$\mathbf{A} \mathbf{U} = \mathbf{b} \quad (4.17)$$

$$A_{ij} = \int_{\Omega} \nabla \phi_j \cdot \nabla \hat{\phi}_i \, dx \quad (4.18)$$

$$b_i = \int_{\Omega} f \hat{\phi}_i \, dx - \int_{\Gamma_N} g \hat{\phi}_i \, ds \quad (4.19)$$

\mathbf{U} is a vector consisting of components U_i for $j = 1, 2, \dots, N$, \mathbf{A} a matrix with the dimensions $N \times N$, and \mathbf{b} is a vector. This problem is solved efficiently by a large variety of numerical methods. We will look into this in section 4.2.

4.1.4 Finite Elements

Finite element methods use piecewise polynomials to approximate solutions. A set of finite elements are introduced to define the approximation space and their basis functions. The finite element definition by Ciarlet [6] states:

Definition 4.1.3. (Finite elements [[6], Chapter 2]) A finite element is a triple $(T, \mathcal{V}, \mathcal{L})$, where

- The domain T is a bounded, closed subset of \mathbb{R}^d (for $d=1,2,3$) with nonempty interior and piecewise smooth boundary.
- The space $\mathcal{V} = \mathcal{V}(T)$ is a finite dimensional function space on T of dimension n .
- The set of degrees of freedom $\mathcal{L} = \{l_1, l_2, \dots, l_n\}$ is a basis for the dual space \mathcal{V}' , that is, the space of bounded linear functionals on \mathcal{V} .

T is a line, triangle or tetrahedron. \mathcal{V} is the second-degree polynomials on T . \mathcal{L} is point evaluation at the vertices and edge midpoints. Introducing a nodal basis for \mathcal{V} will simplify the way we are expressing finite element solutions in \mathbf{V}_h . This is done in the form of basis functions for the local function space \mathcal{V} :

Definition 4.1.4. (Nodal basis [[6], Chapter 2]) The nodal basis $\{\phi_1, \phi_2, \dots, \phi_n\}$ for a finite element $(T, \mathcal{V}, \mathcal{L})$ is the unique basis satisfying:

$$l_i(\phi_j) = \delta_{ij}, \quad i, j = 1, 2, \dots, n \quad (4.20)$$

Finite elements determine the shape of polynomial approximation over each cell, and how those are connected at the cell boundaries. Lagrange elements are one of the most common choices of elements. For the Lagrange elements (Taylor-Hood elements), basis functions are generated by continuous, piecewise polynomial approximations. The solution domain consists of non-overlapping cells, merged into a mesh. The Taylor-Hood elements are used for discretization in this thesis. The cells used to generate the mesh is quite simple in the case of Lagrange elements, i.e., triangles in a two-dimensional mesh, tetrahedrons in 3D, etc. The nodes determine the degrees of freedom through a set of spatial points.

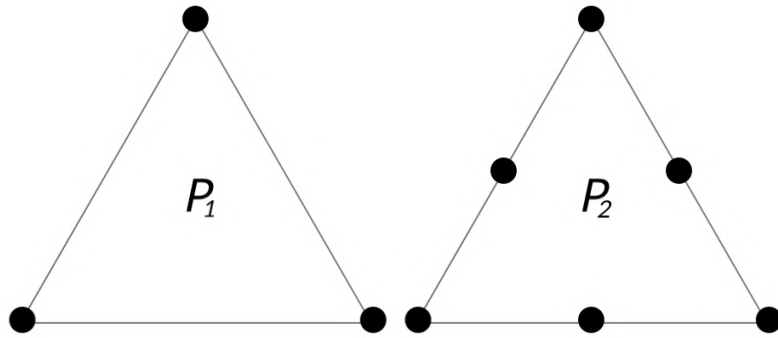


Figure 4.1.1: An illustration of the degrees of freedom for two Taylor-Hood elements in 2D. Linear element (left image) denoted with P_1 , for the pressure p and quadratic element (right image) denoted with P_2 , for the displacement \mathbf{u} .

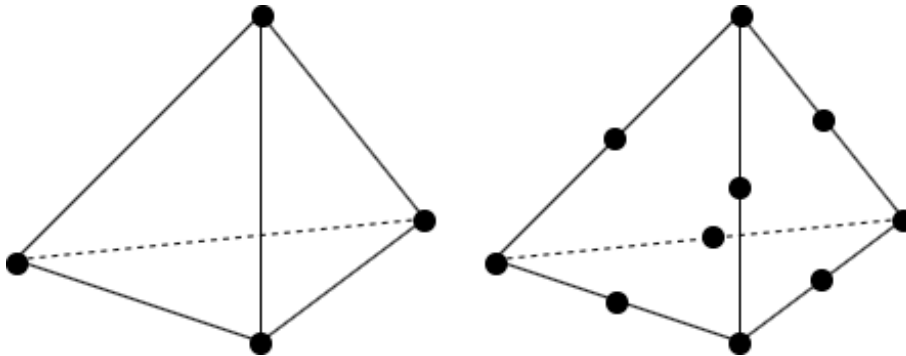


Figure 4.1.2: An illustration of the degrees of freedom for two Taylor-Hood elements in 3D. Linear element (left image) for the pressure p and quadratic conforming element (right image) for the displacements \mathbf{u} on a tetrahedron.

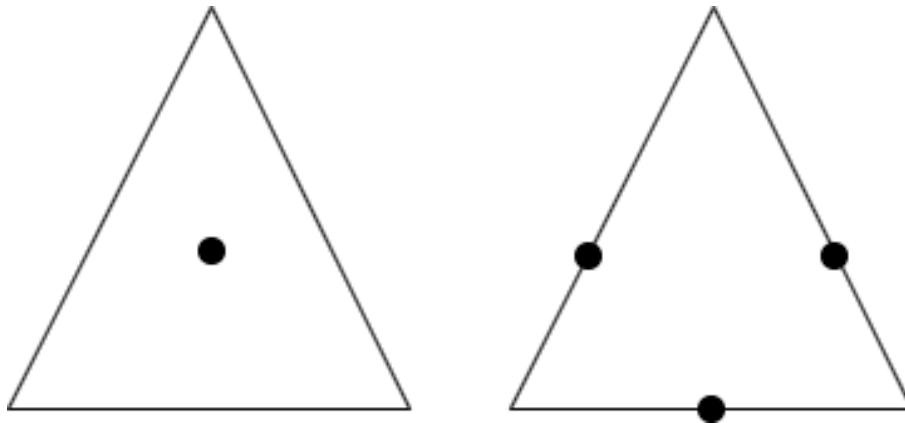


Figure 4.1.3: Crouzeix-Raviart element in 2D. Constant element (left image) for the pressure p and mid-point linear element (right image) for the displacement of u . The displacement is continuous on the mid-points of each element side, while the constant element is discontinuous.

4.2 FEniCS

FEniCS [60] is a modern computing platform with the common goal to enable automated solutions of partial differential equations (PDEs) by the finite element method (FEM). In this thesis, we will use FEniCS version 2019.1.0; hence code implemented may need some changes to work with other versions. This section will start with basic usage and understanding of FEniCS with the implementation of the Poisson problem, before advancing to the coupled Darcy-Stokes problem.

4.2.1 Basic FEniCS Usage

Some preliminaries needs to be clarified before implementing the Poisson problem. Poisson problem was defined thoroughly with the usage of the finite element method in section 4.1, which will be reused in this example with some modifications. The weak formulation:

$$\int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx = \int_{\Omega} f \mathbf{v} \, dx - \int_{\Gamma_N} \frac{\partial \mathbf{u}}{\partial n} \mathbf{v} \, ds \quad (4.21)$$

The Neumann boundary condition Γ_N is obtained from eq. (4.3), where $\frac{\partial \mathbf{u}}{\partial n}$ is the derivative of \mathbf{u} in the outward normal direction at the boundary.

We assume that the test function \mathbf{v} is equal to zero in the entire boundary Γ . This is explained thoroughly in the book by Langtangen and Mardal [25]. The removal of the test function at the boundary will result in following weak formulation for the Poisson problem:

$$\int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx = \int_{\Omega} f \mathbf{v} \, dx \quad (4.22)$$

Next step is to rewrite the problem to a bilinear form, $a(\mathbf{u}, \mathbf{v}) = L(\mathbf{v})$ for all $\mathbf{v} \in \hat{V}$, where we need to find $\mathbf{u} \in V$. The bilinear form and linear form are represented as:

$$a(u, v) = \int_{\Omega} \nabla \mathbf{u} \cdot \nabla \mathbf{v} \, dx \quad (4.23)$$

$$L(v) = \int_{\Omega} f \mathbf{v} \, dx \quad (4.24)$$

We will consider the following definitions for our problem:

- A 2D problem.
- A simple unit square domain $\Omega = [0, 1] \times [0, 1]$.
- An exact solution: $u_0 = 1 + x + 2y^2$.

- Inserting the exact solution into the Poisson problem illustrated in eq. (4.1) provides us with the following source term:

$$- f(x, y) = -6$$

$$- u(x, y) = u_0 = 1 + x + 2y^2$$

The method of manufactured solutions, have been used for calculation of the source term. This method is described thorough in section 5.2.

4.2.2 The Complete Fenics Program

The full FEniCS program for solving the Poisson equation in 2D, is introduced in listing 4.1. The code will be explained thoroughly in this section.

Before we can start using the FEniCS problem-solving environment, we need to import the DOLFIN library:

```
from dolfin import *
```

The next step is to define a mesh. FEniCS have built-in meshes that can be used. In this problem we will be using a very standard uniform mesh, unit square mesh on $[0, 1] \times [0, 1]$, listed as `UnitSquareMesh()`. The arguments for the function are the number of cells in horizontal and vertical directions, where each square divides into two triangles, 20×20 squares will be used in this problem. The last argument in `UnitSquareMesh()` function is optional which indicates the direction of the diagonals, the possibilities are 'crossed,' 'left 'right,' 'left/right' and 'right/left,' but we will discard this argument, and the default direction 'right' will be used.

```
mesh = UnitSquareMesh(20, 20)
```

We define a `FunctionSpace()`, represents a finite element function space. The mesh is implemented as the first argument, the next argument specifies the element family, and the degree of the element is inserted as the last argument. Linear *Lagrange* element family are used for this problem.

```
V = FunctionSpace(mesh, 'Lagrange', 1)
```

Defining the Dirchlet boundary $u_0 = 1 + x + 2y^2$, as follow:

```
u0 = Expression('1 + x[0]*x[0] + 2*x[1]*x[1]', degree=2)
```

The expression is written with C++ syntax, where the independent variables (coordinates) x , y , and z are represented as $x[0]$, $x[1]$, and $x[2]$. An extra argument is needed to specify the dimension of the expression; in our case, this argument is set to `degree = 2`, since we have a 2D problem.

There are two ways to implement the boundary conditions:

```
def boundary(x, on_boundary):  
    return on_boundary
```

The function `boundary` will be used to tell the program where the boundary condition `u0` yield. The function will return `True` or `False` depending on if the given function lies on the Dirichlet boundary or not. This function will be called for every discrete point in the mesh. In our case, we will return `True` in every point around the mesh.

Another way to implement the boundary conditions is by introducing the `near` function.

```
def boundary(x):  
    return (near(x[0], 0.0) or (near(x[0], 1.0)) or  
           (near(x[1], 0.0) or (near(x[1], 1.0)))
```

The function `near` will secure that rounding-off errors do not occur, with a small epsilon as the third argument (`near(x, x0, eps=3e-16)`). Now the Dirichlet boundary condition can be created using the `DirichletBC`:

```
bc = DirichletBC(V, u0, boundary)
```

Where the Function space, boundary condition, and where we assume that the boundary condition applies, are used as arguments.

The next step is to define the trial u and v test function in the function space V . This is defined by a `TrialFunction` and a `TestFunction` on the Function-Space V . Furthermore, the variational problem is defined using FEniCS Unified Form Language (UFL), which is a domain-specific language for declaration of finite element discretizations of variational forms. Hence we can implement the bilinear form a and the linear form L .

```
u = TrialFunction(V)
v = TestFunction(V)
f = Constant(-6.0)
a = inner(grad(u), grad(v))*dx
L = f * v * dx
```

The syntax can be compared to eqs. (4.23) to (4.24). Before we can compute the solution, we need to define a **Function** **u** (a function that lives in a finite element function space) to represent the solution:

```
u = Function(V)
```

Since we have defined **a**, **L**, **u**, and the boundary conditions in **bc**, we can solve the problem with the **solve** function. The solutions will be saved to specified **Function** **u**. Furthermore, we can save the solution to *Visualization Toolkit* (VTK) format such that software like ParaView [18] can be used for a quick visualization of the solution.

```
solve(a == L, u, bc)
file = File("poisson.pvd")
file << u
```

Now that we have a good understanding of how the program is produced, we can start executing the program and examine some different plots:

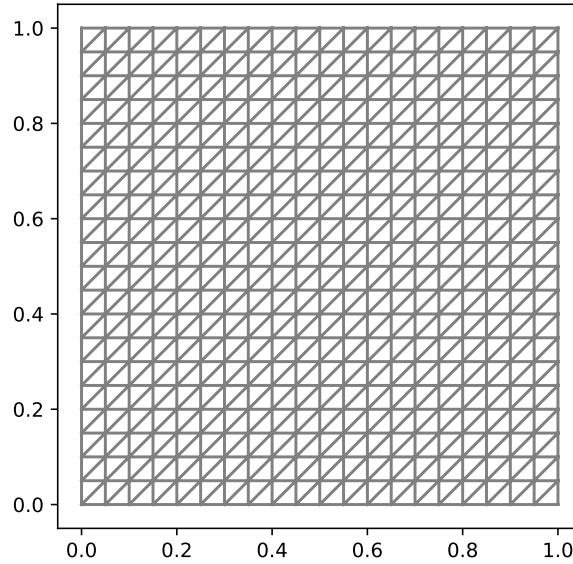


Figure 4.2.4: The generated mesh with the Lagrange elements used for the Poisson equation.

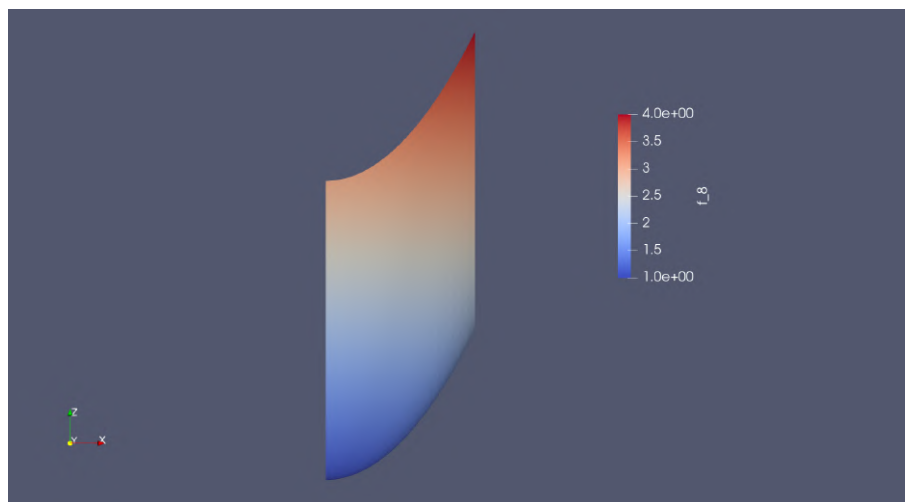


Figure 4.2.5: The solution for u in this basic Poisson example looks as follows. The scaling bar indicates the velocity throughout the domain.

Listing 4.1: Implementasion of Poisson problem in FEniCS

```
from dolfin import *
# Creating the mesh
```



```

mesh = UnitSquareMesh(20, 20)

#Defining the function space
V = FunctionSpace(mesh, 'Lagrange', 1)

# Defining the boundary conditions
u0 = Expression('1 + x[0]*x[0] + 2*x[1]*x[1]', degree=2)

def boundary(x, on_boundary):
    return on_boundary

bc = DirichletBC(V, u0, boundary)

# Define variational problem
u = TrialFunction(V)
v = TestFunction(V)
f = Constant(-6.0)
a = inner(grad(u), grad(v))*dx
L = f * v * dx

# Computing the solution
u = Function(V)
solve(a == L, u, bc)

# Saving the solution to a VTK format for ParaView
file = File("poisson.pvd")
file << u

```

4.3 Advanced FEniCS

The next step is to proceed with the FEM and implementation of coupled Darcy-Stokes equation. A more advanced FEniCS process is introduced for this.

4.3.1 Mesh Generating Software

In this section, we will introduce a mesh generating software Gmsh [15]. This is a three-dimensional finite element mesh generator, which includes advanced visualization techniques. The domain that we are going to generate will imitate the images described in section 2.4, in 2D. Figure 4.3.7 is generated in Gmsh with elementary entities and physical groups. The size of the domain is set to $100 \times 100 \mu\text{m}$.

The next step is to convert the generated mesh into FEniCS for implementation with the coupled problem. The Python script DOLFIN-CONVERT will be used for converting a mesh file from *msh* format to an *xml* format, which is suitable for FEniCS, This script is created by Logg [28]. This is executed with a simple command:

```
> dolfin-convert gmsh_file.msh xml_file.xml
```

which converts a *.msh* file into three *.xml* files:

- *xml_file.xml* which includes all the info of the Mesh.
- *xml_file_facet_region.xml* which includes all the info of the boundaries in the Mesh.
- *xml_file_physical_region.xml* which includes all the info of the domains in the Mesh.

Another way to convert *.msh* files are using the *msh2xdmf.py* program made by Loiseau [31], where meshio is used, which is a python package for smooth reading and converting between generated mesh files. With this program, the *.msh* file will be converted to a more compact file *.xdmf*. The generated file can be used in serial and parallel runs. The execution of this is straightforward:

```
> python3 msh2xdmf.py -d 2 mesh.msh
+-----+
|   GMSH label   | MeshFunction value |
+-----+-----+
| right_boundary |          5          |
| bottom_boundary|          6          |
| left_boundary  |          7          |
| top_boundary   |          8          |
| v_inflow      |          9          |
| v_outflow     |         10          |
| interface_boundary|        11          |
| wall_domain   |          1          |
| p_domain      |          2          |
| v_domain      |          3          |
| interface_domain|          4          |
+-----+-----+
```

This will output the MeshFunction values, used for masking in FEniCS. The connection between the mask and the mesh is illustrated in fig. 4.3.7.

The correlation between the different converting methods can be tested with a simple timing test for a small mesh, for serial and parallel runs. Note that the mesh size used for time testing is minimal, while a larger and more complex mesh will be used for the final computation, where each run will take up to several hours or days.

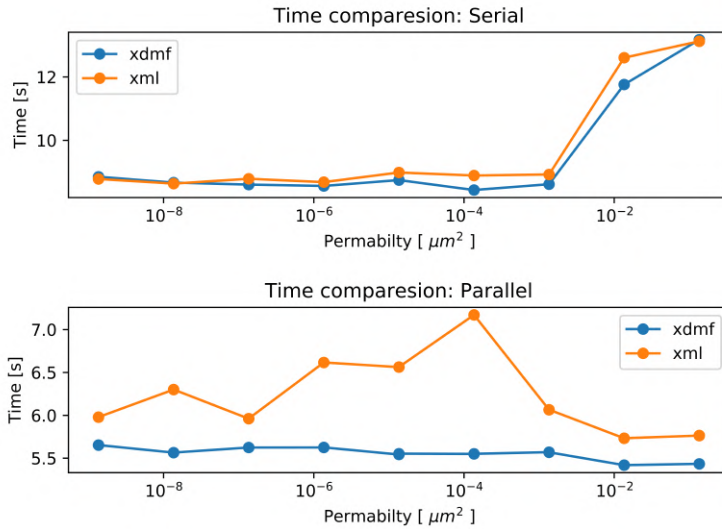


Figure 4.3.6: Serial and parallel time usage for different permeability values. The parallel run is executed with two processors. The different converting methods, *.xml* and *.xdmf*, are shown in the figure.

We will use the converting method by Loiseau [31] for large meshes, with parallel runs. The converting method by Logg [28] will be used for small testing meshes, with serial runs.

The domain generated with Gmsh, will separate into the porous and viscous domains, with an interface domain defined as the blood vessel wall. The boundary conditions shown in fig. 3.3.2 will also be set in Gmsh. This results in a nicely generated mesh, which can be transformed to FEniCS for numerical implementation. FEniCS provides a variety of possibilities when it comes to testing different boundary conditions. The different domains shown in fig. 4.3.7 are represented with MeshFunction value from 1 to 4.

1. The viscous domain where Stokes flow will be implemented.
2. The porous domain where Darcy flow will be implemented.
3. The wall domain which will be used for stopping the unwanted eruption.
4. Interface domain, where wanted eruption will occur.

When it comes to the boundaries, they have been added for the porous and viscous domain. The boundaries around the porous domains are marked as 5 to 8. The inflow and outflow boundaries for the viscous domain are marked as

9 and 10.

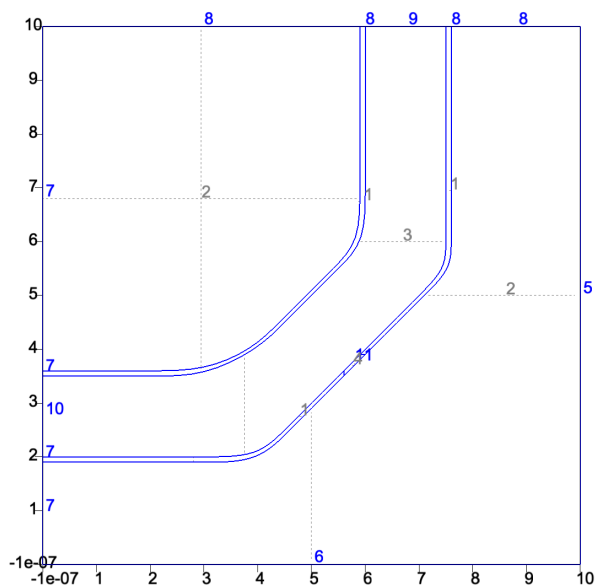


Figure 4.3.7: An illustration of the 2D domain that has been generated with Gmsh and will be used as the simulation domain. The physical surfaces are given as 1 - 4, and the physical curves are given as 5 - 11.

The implementation of the mesh and the splitting of the physical surface and curve are imported to FEniCS by:

```
mesh = Mesh('xml_file.xml')
sub_domains = MeshFunction("size_t", mesh,
    "xml_file_physical_region.xml")
boundaries = MeshFunction("size_t", mesh,
    "xml_file_facet_region.xml")

mark = { "v_domain": 3,
    "p_domain": 2,
    "v_inflow": 9,
    "v_outflow": 10,
    "right_boundary": 5,
    "bottom_boundary": 6,
    "left_boundary": 7,
    "top_boundary": 8,
    "interface_domain": 4,
    "wall_domain": 1,
    "interface_boundary": 11,
```

```
}
```

Marked boundaries/domains associates with the numbers shown in fig. 4.3.7. The marking will be used throughout the code as a definition of the parts in the mesh.

4.3.2 Geometry Storing

The measurements in FEniCS is predefined by `dx`, `ds`, and `dS`, which represents the integration over cells, exterior facets, and interior facets, respectively. When we want to work on different subdomains, integer labels for the different domains or boundaries will be used as arguments. A mapping of the geometry information needs to be stored in the mesh function. This is accomplished with:

```
dx = dx(domain = mesh, subdomain_data=sub_domains)
ds = ds(domain = mesh, subdomain_data=boundaries)
```

For instance, if we want to calculate an equation for the *v_domain*, following code will be implemented:

```
a = inner(u,v)*dx(mark["v_domain"])
```

4.3.3 Dirichlet Boundaries

To specify the Dirichlet boundary conditions for the mesh, we can use mesh functions over each exterior part. An example for implementation of the Dirichlet boundaries, using the marks specified in section 4.3.1:

```
inflow = Expression(("0", "(6-x[0])*(7.5-x[0])"), degree=2)
bc0 = DirichletBC(W.sub(0), inflow, boundaries, mark["v_inflow"])

noslip = Constant((0, 0))
bc1 = DirichletBC(W.sub(0), noslip, boundaries, mark["top_boundary"])
bc2 = DirichletBC(W.sub(0), noslip, boundaries, mark["right_boundary"])
bc3 = DirichletBC(W.sub(0), noslip, boundaries, mark["left_boundary"])
bc4 = DirichletBC(W.sub(0), noslip, boundaries, mark["bottom_boundary"])
```

The inflow has been chosen such that we have a parabolic Poiseuille flow at the right spot. The `Expression` is implemented for the inflow, and no-slip conditions for the boundaries. To collect the boundary conditions, we will use:

```
bc = [bc0, bc1, bc2, bc3, bc4]
```

4.3.4 Mixed Elements

Next, we need to define the function spaces. They are defined separately, before combined into a mixed function space W :

```
V = VectorElement("Lagrange", mesh.ufl_cell(), 2)
Q = FiniteElement("Lagrange", mesh.ufl_cell(), 1)
TH = V * Q
W = FunctionSpace(mesh, TH)
```

The first argument to `FunctionSpace` specifies the type of finite element family. The Taylor-Hood elements are chosen in this thesis. The next step is to define the trial and test functions.

```
(u, p) = TrialFunctions(W)
(v, q) = TestFunctions(W)
```

We have now generated the foundation of the code. The bilinear form a is specified in eqs. (4.38) to (4.40) and the linear form L is specified in eq. (4.41). After these are implemented, a storing function are introduced with the function `Function`, and the actual computation are executed with `solve`.

```
w = Function(W)
solve(a == L, w, bcs)
```

The last step is to split up the velocity and pressure in the solution storing function w , before saving for future analysis in ParaView.

```
(u, p) = w.split(True)
ufile_pvd = File("velocity.pvd")
ufile_pvd << u
pfile_pvd = File("pressure.pvd")
pfile_pvd << p
```

This will generate four new files: *velocity.pvd*, *velocity000000.vtu*, *pressure.pvd* and *pressure000000.vtu*, which can be imported to ParaView.

4.3.5 ParaView

ParaView is a multi-platform data analysis and a visualization application. ParaView can be used to build a visualization of the data using approximation techniques. This will be used throughout the thesis for the visualization of the generated solutions.

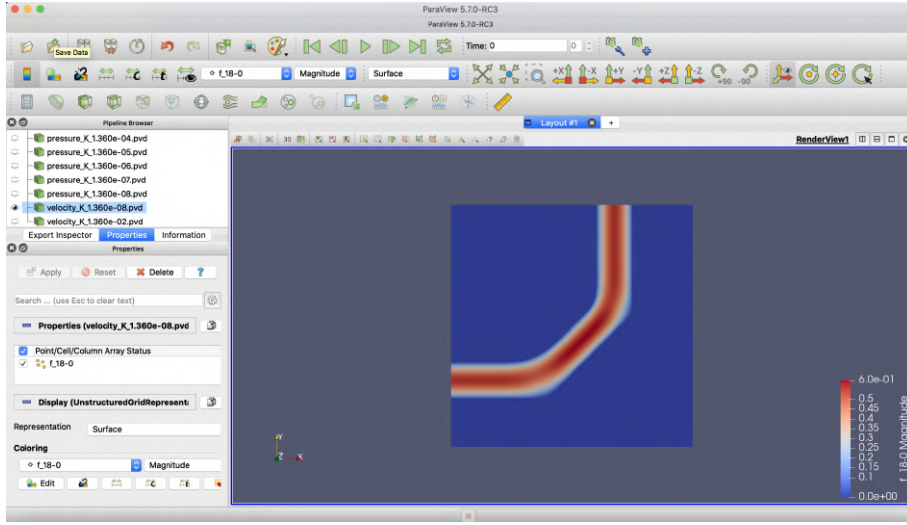


Figure 4.3.8: An illustration of the velocity field in ParaView. The scaling bar shows the velocity through the domain.

ParaView will provide a detailed visualization of the computation executed in FEniCS. Different techniques and method provided by ParaView, makes the visualization and understanding of the generated solutions in FEniCS much easier.

4.4 Unified Discretizations of coupled Darcy–Stokes flow

We have described coupled Darcy-Stokes flow in eq. (3.16) and 3.18. In this section, we will introduce a discretization method for the coupled Darcy-Stokes equation using the finite element method with ideas by Karper et al. [21]. Two weak formulations are introduced in this paper, $H(\text{div})$ -formulation and L^2 -formulation. In this thesis, we will use the L^2 -formulation.

4.4.1 The L^2 -Formulation

In the L^2 -formulation, we define the function spaces for velocity and pressure as \mathbf{V}^2 and Q^2 . When we use this alternative formulation, all the interface conditions represented in eqs. (3.20) to (3.22) are expressed weakly. The subscript s and d represents Stokes and Darcy. The function spaces are described as following:

$$\mathbf{V}^2 = \mathbf{V}^2(\Omega) = \{v \in \mathbf{L}^2 : v_{s,d} \in \mathbf{H}^1(\Omega_{s,d})\} \quad (4.25)$$

$$Q^2 = Q^2(\Omega) = \{q \in L_0^2 : q_{s,d} \in H^1(\Omega_{s,d})\} \quad (4.26)$$

with the corresponding norms:

$$\|\mathbf{u}\|_{\mathbf{V}^2(\Omega)}^2 = \|\mathbf{u}\|_{L^2(\Omega)}^2 + \|\nabla \mathbf{u}\|_{L^2(\Omega_{s,d})}^2 \quad (4.27)$$

$$\|q\|_{Q^2(\Omega)}^2 = \|q\|_{L^2(\Omega)}^2 + \|\nabla q\|_{L^2(\Omega_{s,d})}^2 \quad (4.28)$$

This formulation will be discretized using Taylor-Hood elements, described in section 4.1.4. When using Taylor-Hood elements, we will get a stable mixed problem, where we have P_2 elements for the velocity and P_1 elements for the pressure. These elements are shown in fig. 4.1.1. Taylor-Hood elements results in a discrete function spaces $\mathbf{V}_h \in \mathbf{V}^2$ and $Q_h \in Q^2$. Furthermore, the elements are conforming, which states that the edges and faces of neighboring elements match exactly.

Inner Product Notation

Inner products will be denoted with $\langle \cdot, \cdot \rangle$. The L^2 inner product of \mathbf{u} and \mathbf{v} , where \mathbf{u} and \mathbf{v} is either scalar function or vector-valued, on a measure space Ω , is defined as:

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\Omega} = \int_{\Omega} \mathbf{u} \cdot \mathbf{v} \, d\Omega \quad (4.29)$$

Unified Variational Formulation

The goal of this section is to formulate the L^2 -formulation of the coupled Stokes-Darcy equation and find the functions $(\mathbf{u}, p) \in \mathbf{V}^2 \times Q^2$ such that for all $(\mathbf{v}, q) \in \hat{\mathbf{V}}^2 \times \hat{Q}^2$:

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) + b(p, \mathbf{v}) &= l_1(\mathbf{v}) \\ b(q, \mathbf{u}) &= 0 \end{aligned} \quad (4.30)$$

$a(\cdot, \cdot)$ and $b(\cdot, \cdot)$ represents the bilinear forms, while $l_1(\cdot)$ represents the linear form.

We will start by multiplying the momentum equation eq. (3.5) and Darcy equations (3.11), with the test function \mathbf{v} . q will be multiplied with the continuity equations defined in eq. (3.6) and (3.14).

Stokes and Darcy domains will be represented as Ω_s and Ω_d . The equations represented in (3.5) - (3.8) for Stokes system and (3.11) - (3.14) for Darcy system will be used for the calculations of the unified variational formulation, with the following interface condition:

$$\mathbf{u}_s \cdot \mathbf{n}_s = \mathbf{u}_d \cdot \mathbf{n}_d \quad \text{on } \Gamma \quad (4.31)$$

$$2\mu \mathbf{n}_s \cdot \epsilon(\mathbf{u}_s) \cdot \mathbf{n}_s = p_s - pd \quad \text{on } \Gamma \quad (4.32)$$

$$2\mathbf{n}_s \cdot \epsilon(\mathbf{u}_s) \cdot \boldsymbol{\tau} = \alpha K^{-\frac{1}{2}} \mathbf{u}_s \cdot \boldsymbol{\tau} \quad \text{on } \Gamma \quad (4.33)$$

The next step is to integrate the equations and represented the integrals with the L^2 -inner product notation over the entire domain Ω .

$$\begin{aligned} -2\mu\langle\nabla\cdot\epsilon(\mathbf{u}),\mathbf{v}\rangle_{\Omega_s}+\langle\nabla p,\mathbf{v}\rangle_{\Omega_s}&=\langle\mathbf{f},\mathbf{v}\rangle_{\Omega_s} \\ \langle\nabla\cdot\mathbf{u},q\rangle_{\Omega_s}&=0 \\ \frac{\mu}{K}\langle\mathbf{u},\mathbf{v}\rangle_{\Omega_d}-\langle\nabla p,\mathbf{v}\rangle_{\Omega_d}&=\langle f,\mathbf{v}\rangle_{\Omega_d} \\ \langle\nabla\cdot\mathbf{u},q\rangle_{\Omega_d}&=0 \end{aligned}$$

Now that we have rearranged the equations, we will combine them, such that we can view the problem as coupled Darcy-Stokes:

$$\begin{aligned} -2\mu\langle\nabla\cdot\epsilon(\mathbf{u}),\mathbf{v}\rangle_{\Omega_s}+\frac{\mu}{K}\langle\mathbf{u},\mathbf{v}\rangle_{\Omega_d}+\langle\nabla p,\mathbf{v}\rangle_{\Omega}&=\langle\mathbf{f},\mathbf{v}\rangle_{\Omega} \\ \langle\nabla\cdot\mathbf{u},q\rangle_{\Omega}&=0 \end{aligned}$$

Integrating by part will be calculated for almost each term. We will go through each part step by step:

$$\begin{aligned} -2\mu\langle\nabla\cdot\epsilon(\mathbf{u}),\mathbf{v}\rangle_{\Omega_s}&=2\mu\langle\epsilon(\mathbf{u}),\epsilon(\mathbf{v})\rangle_{\Omega_s}-2\mu\langle\mathbf{n}_s\cdot\epsilon(\mathbf{u})\mathbf{n}_s,\mathbf{v}\cdot\mathbf{n}_s\rangle_{\Gamma_s} \\ &-2\mu\langle\mathbf{n}_s\cdot\epsilon(\mathbf{u})\cdot\boldsymbol{\tau},\mathbf{v}\cdot\boldsymbol{\tau}\rangle_{\Gamma_s} \\ &=2\mu\langle\epsilon(\mathbf{u}),\epsilon(\mathbf{v})\rangle_{\Omega_s}-\langle p_s-p_d,\mathbf{v}\cdot\mathbf{n}_s\rangle_{\Gamma_s} \\ &-\mu\langle\alpha K^{\frac{1}{2}}\mathbf{u}\cdot\boldsymbol{\tau},\mathbf{v}\cdot\boldsymbol{\tau}\rangle_{\Gamma_s} \end{aligned} \tag{4.34}$$

$$\tag{4.35}$$

The interface conditions (4.32) and (4.33) have been inserted for second and third term, which is reviewed in the article by Layton et al. [26]. The second term in eq. (4.35) will cancel out by using the Taylor-Hood elements. The results from Drøsdal [10] concludes that the Beavers-Joseph-Saffman condition (4.35) can be removed from the variational formulation. This simplification will give an advantage when working on a complex geometry, as it is challenging to implement the tangent vector in FEniCS.

Given the following notation:

$$\nabla\mathbf{u}=\sum_{ij}\int_{\Omega}\frac{\partial u_i}{\partial x_j}d\Omega, \quad (\nabla\mathbf{u})^T=\sum_{ij}\int_{\Omega}\frac{\partial u_j}{\partial x_i}d\Omega$$

The equivalence between $\langle \epsilon(\mathbf{u}), \epsilon(\mathbf{v}) \rangle$ and $\langle \epsilon(\mathbf{u}), \nabla \mathbf{v} \rangle$ are given as following:

$$\begin{aligned}
\langle \epsilon(u), \epsilon(v) \rangle &= \sum_{ij} \int_{\Omega} \frac{1}{4} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) d\Omega \\
&= \frac{1}{4} \sum_{ij} \int_{\Omega} \left[\frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} + \frac{\partial u_i}{\partial x_j} \frac{\partial v_j}{\partial x_i} + \frac{\partial u_j}{\partial x_i} \frac{\partial v_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \frac{\partial v_j}{\partial x_i} \right] d\Omega \\
&= \frac{1}{4} \left[\sum_{ij} \int_{\Omega} \frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} d\Omega + \sum_{ij} \int_{\Omega} \frac{\partial u_j}{\partial x_i} \frac{\partial v_i}{\partial x_j} d\Omega \right. \\
&\quad \left. + \sum_{ij} \int_{\Omega} \frac{\partial u_j}{\partial x_i} \frac{\partial v_i}{\partial x_j} d\Omega + \sum_{ij} \int_{\Omega} \frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} d\Omega \right] \\
&= \frac{2}{4} \left[\sum_{ij} \int_{\Omega} \frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} d\Omega + \sum_{ij} \int_{\Omega} \frac{\partial u_j}{\partial x_i} \frac{\partial v_i}{\partial x_j} d\Omega \right] \\
&= \frac{1}{2} \left(\sum_{ij} \int_{\Omega} \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \frac{\partial v_i}{\partial x_j} d\Omega = \langle \epsilon(u), \nabla v \rangle_{\Omega}
\end{aligned}$$

Integrating by parts on the terms that contains pressure:

$$\langle \nabla p, \mathbf{v} \rangle_{\Omega} = \langle \nabla p, \mathbf{v} \rangle_{\Omega_d} - \langle p, \nabla \cdot \mathbf{v} \rangle_{\Omega_s} + \langle p, \mathbf{v} \cdot \mathbf{n}_s \rangle_{\Gamma_{\partial s}} \quad (4.36)$$

To achieve symmetry, we will replace $\hat{p} = -p$ for the pressure in the Stokes domain, and \hat{p} is presented as p throughout the thesis. The only term we have left now is the equations of continuity for each flow:

$$\langle \nabla \cdot \mathbf{u}, q \rangle_{\Omega} = \langle \nabla \cdot \mathbf{u}, q \rangle_{\Omega_d} + \langle \nabla \cdot \mathbf{u}, q \rangle_{\Omega_s} \quad (4.37)$$

The considerations above give rise to the variational formulation given at equation (4.30), with the bilinear forms $\mathbf{a}(\mathbf{u}, \mathbf{v})$, $b(p, \mathbf{v})$ and $b(q, \mathbf{u})$ is given as:

$$\mathbf{a}(\mathbf{u}, \mathbf{v}) = 2\mu \langle \epsilon(\mathbf{u}), \epsilon(\mathbf{v}) \rangle_{\Omega_s} + \frac{\mu}{K} \langle \mathbf{u}, \mathbf{v} \rangle_{\Omega_d} \quad (4.38)$$

$$b(p, \mathbf{v}) = \langle \nabla p, \mathbf{v} \rangle_{\Omega_d} + \langle p, \nabla \cdot \mathbf{v} \rangle_{\Omega_s} + \langle p, \mathbf{v} \cdot \mathbf{n}_s \rangle_{\Gamma_{\partial s}} \quad (4.39)$$

$$b(q, \mathbf{u}) = \langle \nabla \cdot \mathbf{u}, q \rangle_{\Omega_d} + \langle \nabla \cdot \mathbf{u}, q \rangle_{\Omega_s} \quad (4.40)$$

and the linear form $l_1(\mathbf{v})$ is given as:

$$l_1(\mathbf{v}) = \langle \mathbf{f}, \mathbf{v} \rangle_{\Omega_s} + \langle \mathbf{f}, \mathbf{v} \rangle_{\Omega_d} \quad (4.41)$$

Collecting the forms in the coupled Darcy-Stokes formulation, gives us the variational formulation: Find $(\mathbf{u}, p) \in \mathbf{V}^2 \times Q^2$, such that

$$\mathbf{a}(\mathbf{u}, \mathbf{v}) + b(p, \mathbf{v}) = l_1(\mathbf{v}) \quad (4.42)$$

$$b(q, \mathbf{u}) = 0 \quad (4.43)$$

for all $(\mathbf{v}, q) \in \hat{\mathbf{V}}^2 \times \hat{Q}^2$.

The Continuous Case

We will now establish the existence, uniqueness, and stability for saddle point problems on the form eqs. (4.42) to (4.43). We will introduce four conditions, which need to be satisfied in order to have a well-posed problem, i.e., a problem that allows a unique solution. A more extensive explanation of the conditions is shown in the paper by Logg and Mardal [29].

The continuity/boundedness of a :

$$a(\mathbf{u}, \mathbf{v}) \leq C_1 \|\mathbf{u}\|_{\mathbf{V}} \|\mathbf{v}\|_{\mathbf{V}}, \quad \forall \mathbf{u} \in \mathbf{V}, \forall \mathbf{v} \in \mathbf{V} \quad (4.44)$$

The continuity/boundedness of b :

$$b(\mathbf{u}, q) \leq C_2 \|\mathbf{u}\|_{\mathbf{V}} \|q\|_Q, \quad \forall \mathbf{u} \in \mathbf{V}, \forall q \in Q \quad (4.45)$$

The coercivity of a

$$a(\mathbf{u}, \mathbf{u}) \geq C_3 \|\mathbf{u}\|_{\mathbf{V}}^2, \quad \forall \mathbf{u} \in \mathbf{Z} \quad (4.46)$$

Where $\mathbf{Z} = \{\mathbf{u} \in \mathbf{V} \mid b(\mathbf{u}, q) = 0, \quad \forall q \in Q\}$,

The surjectivity of b , also called inf-sup condition:

$$\sup_{\mathbf{u} \in \mathbf{V}} \frac{b(\mathbf{u}, q)}{\|\mathbf{u}\|_{\mathbf{V}}} \geq C_4 \|\mathbf{u}\|_{\mathbf{V}}^2, \quad \forall q \in Q \quad (4.47)$$

The Discrete Case

The next step is to discretize the system (4.30), with the Taylor-Hood element we discussed in section 4.1.4. This results in the discrete saddle point problem: find $\mathbf{u}_h \in \mathbf{V}$ and $p_h \in Q_h$, such that:

$$\begin{aligned} a(\mathbf{u}_h, \mathbf{v}_h) + b(p_h, \mathbf{v}_h) &= l_1(\mathbf{v}_h) \\ b(q_h, \mathbf{u}_h) &= 0 \end{aligned} \quad (4.48)$$

for all $\mathbf{v}_h \in \hat{\mathbf{V}}_h$ and $q_h \in \hat{Q}_h$.

Similar to the continuous case in section 4.4.1, the finite element spaces \mathbf{V}_h and Q_h needs to be satisfied for the conditions shown in eqs. (4.44) to (4.47). These conditions will satisfy for the discretized space, considering that we are working with Taylor-Hood elements. This indicates that the discrete problem is well-posed (has unique solutions) and that the solution converges to the continuous solution when $h \rightarrow 0$. This is explained more thoroughly in the paper by Ellingsrud [11].

A Priori Error Estimates

To examine the error in the finite element, we need to compare the solutions of the system in eq. (4.30) and (4.48). Under the assumption that we have a

stable scheme, a priori error estimate can be obtained:

$$\|\mathbf{u} - \mathbf{u}_h\|_{\mathbf{V}} + \|p - p_h\|_Q \leq C \left(\inf_{\mathbf{v}_h \in \mathbf{V}} \|\mathbf{u} - \mathbf{v}_h\|_{\mathbf{V}} + \inf_{Q_H \in Q} \|p - q_h\|_Q \right) \quad (4.49)$$

We will assume that the velocity is represented with one order higher than the pressure. In Taylor-Hood elements, the velocity is approximated with quadratic polynomial, and the pressure is approximated with a linear function. This will simplify the error estimate of the coupled Darcy-Stokes to:

$$\|\mathbf{u} - \mathbf{u}_h\|_{H^1} + \|p - p_h\|_{L^2} \leq Ch^3 \left(\|\mathbf{u} - \mathbf{v}_h\|_{H^3} + \|p - q_h\|_{H^2} \right) \quad (4.50)$$

This is implemented and tested in the paper by Ellingsrud [11].

Development of Pressure and Velocity

The goal is to analyze the progression of the pressure and velocity change in each experiment with increasing permeability, such that it can be compared to the calculated cases shown in section 2.4.1. This results in an overview of development in pressure and velocity throughout the different choices of permeability executed in chapter 5. Following methods are introduced for studying the development:

$$\int_{\Omega} p_h \, d\Omega \quad (4.51)$$

This will be used for modeling the pressure over a domain, and for the velocity we will use:

$$\int_{\Omega} \mathbf{u}_h \cdot \mathbf{u}_h \, d\Omega \quad (4.52)$$

Those methods of calculations will be implemented in FEniCS, and we will study the changes in each domain for each tested permeability.

4.4.2 Implementation in FEniCS

The FEniCS implementation of the coupled Darcy-Stokes scheme, for the variational form, is included in listing 4.2. The code shows a general way to compute the coupled scheme. Few conditions needs to be noted:

- The domain is divided into four subdomains: viscous, porous, wall, and interface, marked in FEniCS by:
 - Viscous domain: 'v_domain'.
 - Porous domain: 'p_domain'.
 - Interface domain: 'interface_domain'.
 - wall domain: 'wall_domain'.

- The boundaries have also been marked (shown in section 4.3.1).

Listing 4.2) has been used to generate the simulations that we are going to study in chapter 5. Note that the code shown below only holds for the generalized case.

Listing 4.2: The implementation of the variation form of coupled scheme

```

from dolfin import *
mesh = Mesh('mesh.xml')
sub_domains = MeshFunction("size_t", mesh,
                           "mesh_physical_region.xml")
boundaries = MeshFunction("size_t", mesh,
                          "mesh_facet_region.xml")

mark = { "v_domain": 3,
         "p_domain": 2,
         "v_inflow": 9,
         "v_outflow": 10,
         "right_boundary": 5,
         "bottom_boundary": 6,
         "left_boundary": 7,
         "top_boundary": 8,
         "interface_domain": 4,
         "wall_domain": 1,
         }

dx = dx(domain = mesh, subdomain_data=sub_domains)
ds = ds(domain = mesh, subdomain_data=boundaries)

# Function spaces
V = VectorElement("Lagrange", mesh.ufl_cell(), 2)
Q = FiniteElement("Lagrange", mesh.ufl_cell(), 1)
TH = V * Q
W = FunctionSpace(mesh, TH)

# VISCOUS FLOW -- DIRICHLET BOUNDARIES --

# Inflow boundary condition for velocity
inflow = Expression(("0", "(6-x[0])*(7.5-x[0])"), degree=2)
bc1_v = DirichletBC(W.sub(0), inflow,
                   boundaries, mark["v_inflow"])

# Boundary condition for pressure at outflow
p0 = Constant((0.0))
bc2_v = DirichletBC(W.sub(1), p0, boundaries, mark["v_outflow"])

# POROUS FLOW -- Dirichlet BOUNDARIES --
# Boundary conditions
noslip = Constant((0, 0))
bc0_p = DirichletBC(W.sub(0), noslip,
                   boundaries, mark["top_boundary"])
bc1_p = DirichletBC(W.sub(0), noslip,
                   boundaries, mark["right_boundary"])
bc2_p = DirichletBC(W.sub(0), noslip,
                   boundaries, mark["left_boundary"])
bc3_p = DirichletBC(W.sub(0), noslip,
                   boundaries, mark["bottom_boundary"])

```

```

# Collected conditions
bcs = [bc1_v, bc2_v, bc0_p, bc1_p, bc2_p, bc3_p]

# Permeability and viscosity
K_p = Constant(1.36 * 10**(-9))
K_interface = Constant(1.36 * 10**(-9))
K_wall = Constant(1.36 * 10**(-9))
mu = Constant(0.00496 )

def epsilon(u):
    return 0.5*(nabla_grad(u) + nabla_grad(u).T)

f_p = Expression(("0", "mu/K_p*(6-x[0])*(7.5-x[0])"),
    mu = mu, K_p = K_p, degree=2)
f_v = Constant((0, 0))

# Define variational problem
(u, p) = TrialFunctions(W)
(v, q) = TestFunctions(W)

a = (2*mu*inner(epsilon(u), epsilon(v)) + div(v) * p
    + q * div(u)) * dx(mark["v_domain"])
    + ((mu/ K_p) * inner(u, v) + inner(grad(p), v)
    + q * div(u)) * dx(mark["p_domain"])
    + ((mu/ K_wall) * inner(u, v) + inner(grad(p), v)
    + q * div(u)) * dx(mark["wall_domain"])
    + ((mu/ K_interface) * inner(u, v) + inner(grad(p), v)
    + q * div(u)) * dx(mark["interface_domain"])
L = inner(f_p, v) * ds(mark["p_domain"])
    + inner(f_v, v) * ds(mark["v_domain"])

# Compute solution
w = Function(W)
solve(a == L, w, bcs)

# Split the mixed solution using deepcopy
(u, p) = w.split(True)

# Saving the plots for Paraview
ufile_pvd = File("velocity.pvd")
ufile_pvd << u
pfile_pvd = File("pressure.pvd")
pfile_pvd << p

```

Chapter 5

Numerical Experiments

In this chapter, we will edit/configure listing 4.2), with a goal to simulate the experiments described in section 2.4.1, with different criteria.

5.1 Element Scaling

Scaling is an essential part of the grid generation, so that it can be compared to the real-life case. An illustration of the mesh generated in Gmsh, with the different measurements, is shown in fig. 5.1.1. The entire grid is enclosed in a $100 \times 100 \mu m^2$ area. A scaling bar set to $10 \mu m$ is shown in the figure. The sizes calculated from chapter 2, and other measurement used in the implementation are given by:

Table 5.1: A list of measurements used for generating the grid

Permeability (resting condition)	$1.36 \cdot 10^{-9} \mu m^2$
Viscosity	0.00496 Pa·s
Length (viscous domain)	$130 \mu m$
Diameter (viscous domain)	$15 \mu m$
Gap size(interface domain)	$2.8 \mu m$ to $4 \mu m$
Diameter of the wall	$0.6 \mu m$

The chosen permeability for healthy tissues is proposed by Smith and Humphrey [52]. The noted gap size is chosen temporarily. A smaller and more accurate size is also implemented later in section 5.3.3. The measurement for permeability and gap size measurements are not fixed, as we will simulate for increased permeability with different gap sizes.

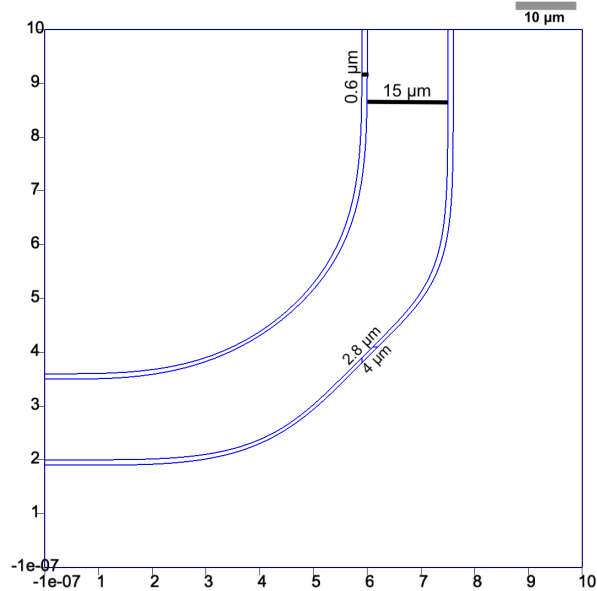


Figure 5.1.1: An illustration of the grid with a scale bar set to $10 \mu m$. The range of the x- and y-axis goes from 0 to 10.

5.1.1 Mesh

Choosing the right mesh size is a challenging part of FEM. Too large finite element size will lead to inaccurate results, but too smaller finite element size becomes computationally expensive. We need to choose an element size that is possible to compute and gives an accurate visualization of the flow.

The domain shown in fig. 4.3.7, has been chosen as the domain for the coupled Darcy-Stokes scheme. The size of the elements has been set using Gmsh. The element distribution in Gmsh are chosen by:

- Computing element sizes using point values.
- Increased amount of element around boundaries.
- Frontal-Delaunay [40] has been used as a two-dimensional algorithm for the calculation of the elements.

This will generate triangles between each point that are specified on the domain.

The variables that are changed to get an optimal element distribution is the element size factor, which scales the triangles to even smaller triangles. The scaling process of a triangles are shown in fig. 5.1.2. The element that has not been scaled have a baseline of around $1 \mu m$ (image a) in fig. 5.1.2).

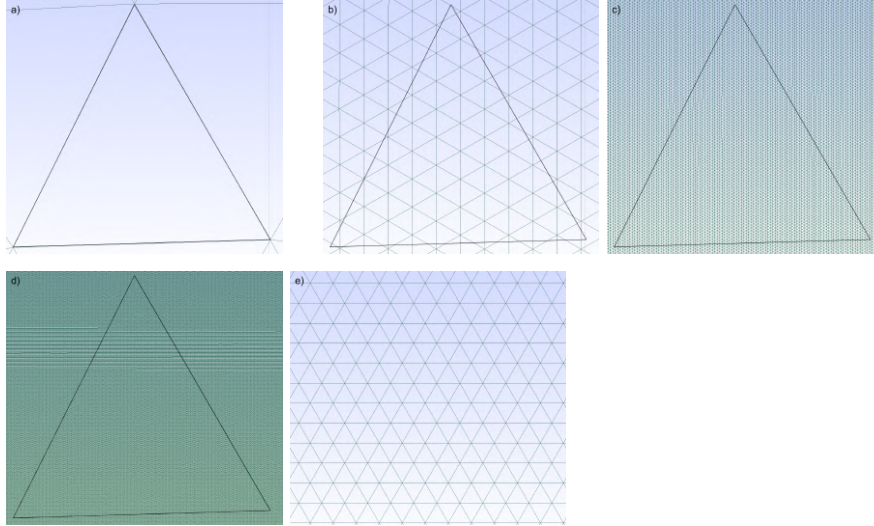


Figure 5.1.2: An illustration of different scaled elements sizes. a) No scaling, with a baseline equal to $1 \mu m$, b) scale 0.1, c) scale 0.01, d) scale 0.004, e) is a close up of a small area of d).

In order to generate a suitable finite element size for our problem, the following tests are performed:

- Execute the analysis for several different finite element sizes.
- Take note of the high deformations, and refine the mesh in these areas.
- Collect the outcomes, number of nodes, computing time, etc. from each test.

The cells and vertices distribution for the different meshes are shown in table 5.2. ParaView is used for analyzing the different sized meshes. The function `plot over line` in ParaView will be used to illustrate the development throughout the meshes. This function generates an diagonal line across the mesh. This is shown in fig. 5.1.3.

Table 5.2: A View over the size of the scaled meshes

Scaling factors	Vertices	Cells
0.009	1 435 536	2 866 617
0.005	4 635 786	9 263 570
0.004	7 248 609	14 487 216

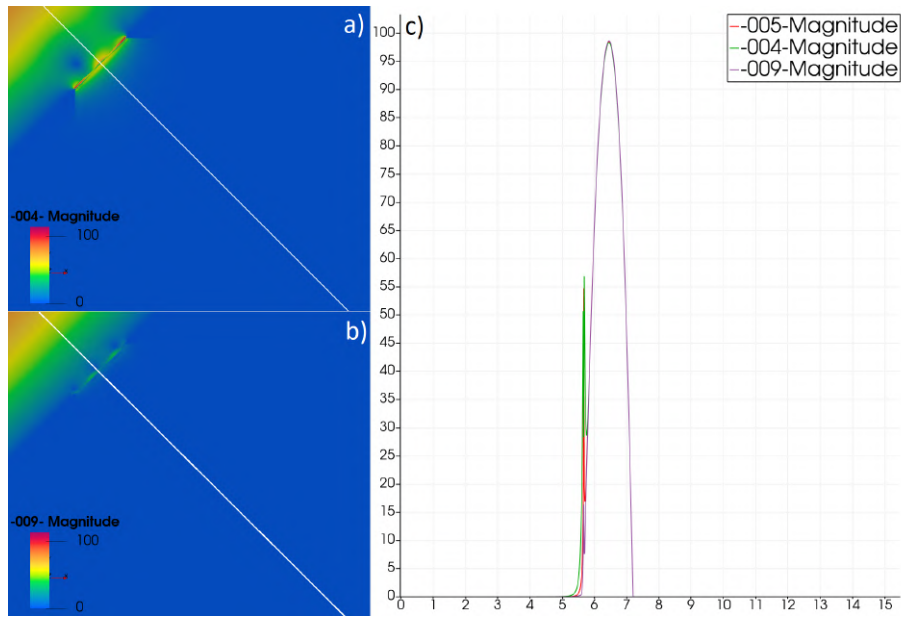


Figure 5.1.3: Scaling factor 0.004 is shown in a), scaling factor 0.09 is shown in b). The white lines in a) and b) illustrate where the velocity development is measured. c) shows the velocity development for scaling factor 0.004, 0.005 and 0.009 along the line (the label indicates which element scale each line represents). The permeability is set to $1.36 \cdot 10^{-5} \mu m^2$ for all testes.

From fig. 5.1.3 a), we see that the smallest element sizes with a scaling of 0.004 provides a good view of the velocity around the compact area. The simulations from the zoomed-in images shown in fig. 5.1.3 a) and b), we see the deformation becomes more detailed as the scaling factor decreases. This choice of element size indicates that we get a lot more information around the compact and small gap area. Generating a finite elements scaled smaller than 0.004, becomes too computationally heavy. Therefore, scaling factor 0.004 will be used for the rest of the simulations. The full mesh is illustrated in fig. 5.1.4.

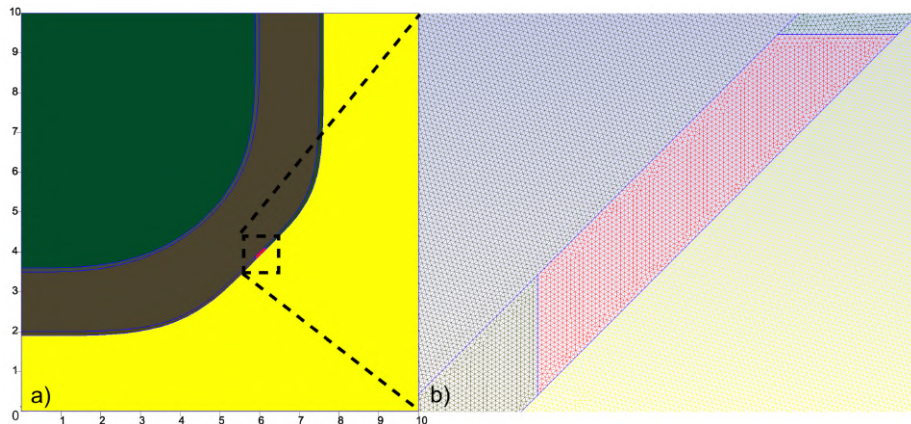


Figure 5.1.4: An overview of the final mesh. a) The entire domain with a finite element scaling of 0.004. b) A closeup view over the size of the elements in the compact area.

5.2 The Method of Manufactured Solutions

The method of manufactured solutions (MMS) is a method for verifying the accuracy of the generated analytical solution in the numerical experiments. The MMS is executed with the following steps:

- Assuming a function is a solution to the problem.
- Plugin the solution function into the equation to get the source and boundary terms.
- Solve the problem with those conditions as a guide for the simulation tool.
- Compare the numerical result with the chosen analytical (manufactured) solution.

The steps are described more detailed in [22],[47] and [37]. The concept of MMS is to use the manufactured solution as an exact solution to a PDE or a set of PDEs, which have been constructed by solving a given problem backward. This is done by fitting the boundary condition and the source terms according to the chosen solution. The source term is then added to the original PDE, such that the assumed function in the solution satisfies the new PDE exactly. The solution that has been generated using the manufactured solution is used for comparison against the numerical solution. The chosen solutions are generated when the parameter h tends to zero in the saddle point problem (4.48). Furthermore, a measurement of the error is obtained. The accuracy of the code is determined by running the problem on refined grids. A verification test using the method of manufactured solutions provides the precise result as to whether or not the

numerical method implemented is correct.

After the manufactured solution is introduced, the source terms are adjusted. f_v and f_p represents the source terms in the momentum equation for the viscous (3.16) and the Darcy's equation for the porous region (3.18). The source term in the continuity equation is zero because of the divergence-free velocity, eq. (3.17), and (3.19). Calculation of the source term is done by inserting the exact solutions in eq. (3.5) and (3.11). The MMS are executed for each numerical experiments, separately, shown in section 5.3.2 and 5.3.3.

5.3 Numerical Experiments

In this section, we will simulate low permeability, increased permeability, and hyperpermeability in a two-dimensional blood flow model. Solutions generated from the numerical method will be used for comparison with the eruptions that are described in section 2.4.1. The goal is to reproduce the images by simulating the stationary problem with increased permeability rather than the non-stationary development. We will also test how accuracy is affected by the physical parameters and boundary conditions.

5.3.1 Simulation Around the Interface

We assume that the burst develops simultaneously with increased permeability. The endothelial cell will contract and lead to vascular leakage into the porous domain with increased permeability. After a while, the flow will be stabilized, and there will be no more drastic changes to the leakage at the interface, this is where we can assume that the cells have closed up. The endothelial cell contraction size and the endothelial cell surrounding the viscous domain is referred to as a gap and a wall, respectively, for the rest of the thesis.

The simulations are shown with scaled velocity vectors that indicate the direction of the stream. The `plot over line` function is introduced for visualization of the velocity development diagonally on the domain. The pressure development is described with a color map.

It is important to remember that we only look at the magnitude of the pressure profile. This will give us a view over how the increasing permeability affects the development of the gap. Each step in the axis that is shown in the simulations represents $10 \mu m$ in x- and y-direction. We have tried to be consecutive with the size of the scale bar and the amount of velocity vectors.

5.3.2 Simulation: Eruption in a Closed Grid

In this section, we will try to simulate how the eruption occurs in the enclosed grid, to replicate the phenomena inside the red rectangle shown in fig. 3.3.3.

Preliminaries

The permeabilities are evenly distributed between $1.36 \cdot 10^{-9} \mu m^2$ and $1.36 \cdot 10^{-1} \mu m^2$ for mimic the development through time, as shown in fig. 2.4.13 chapter 2. The following boundary conditions are used for this simulation:

- v_{in} at the inflow boundary.
- No flow conditions, $u \cdot n = 0$, around the porous domain.
- $p_{out} = 0$ at the outflow boundary.

The flow starts from the top in the viscous domain, represented as v_{in} in fig. 3.3.2 and flow through the vicious domain. These conditions are set such that we can reproduce the real case problem discussed in chapter 2.

The manufactured solutions are given by:

$$\mathbf{u}_e = \begin{pmatrix} 0 \\ (6-x)(7.5-x) \end{pmatrix}, \quad p_e = 6.87 \quad (5.1)$$

$$f_v = -2\mu \nabla \cdot \epsilon(\mathbf{u}_e) + \nabla p = (0, 0) \quad (5.2)$$

$$f_p = \frac{\mu}{\kappa} \mathbf{u}_e - \nabla p = (0, \frac{\mu}{\kappa} (6-x)(7.5-x)) \quad (5.3)$$

The exact solution for pressure p_e is calculated from section 2.2.3 for a $130 \mu m$ long vessel.

The domain is chosen, so the grid is covered by a solid substance, which prevents flow from streaming any further.

Simulation

Now that we have defined the boundary conditions, the method of manufactured solutions, and stated the area we want to reproduce, listing 4.2 can modify. The following modification to the code has been implemented:

```
inflow = Expression(("0", "180*(6-x[0])*(7.5-x[0])"), degree=2)
bc3_v = DirichletBC(W.sub(0), inflow, boundaries_mf, mark["v_inflow"])

noslip = Constant((0, 0))
bc0_p = DirichletBC(W.sub(0), noslip,
                    boundaries_mf, mark["top_boundary"])
bc1_p = DirichletBC(W.sub(0), noslip,
                    boundaries_mf, mark["right_boundary"])
bc2_p = DirichletBC(W.sub(0), noslip,
                    boundaries_mf, mark["left_boundary"])
bc3_p = DirichletBC(W.sub(0), noslip,
                    boundaries_mf, mark["bottom_boundary"])

# Collected conditions
```

```
bcs = [bc3_v, bc0_p, bc1_p, bc2_p, bc3_p]
```

and the source terms:

```
f_p = Expression(("0", "mu/K_p*(6-x[0])*(7.5-x[0])"),
                 mu = mu, K_p = K_p, degree=2)
f_v = Constant((0, -6.87))
```

Case 1: Low Permeability

We need to ensure that we have a resting case, such that the comparison with the fig. 2.4.13, where the time is equal to zero, becomes as accurate as possible. For this to occur, we need to choose low permeability for walls around the viscous domain and in the gap between the viscous and porous domains. This will make sure that the wall is closed up for all bursts. The boundary condition is set so that the flow is enclosed in the $100 \times 100 \mu m^2$ area. These conditions enable us to study the eruption in a small area without any further leakage.

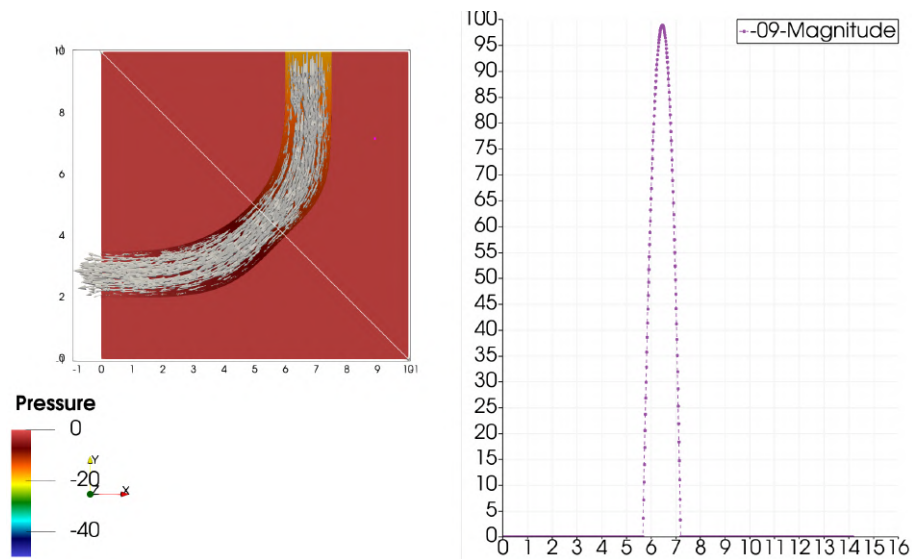


Figure 5.3.5: Left: Simulation of viscous and porous flow in an enclosed mesh with very low permeability. In this case, the gap does not affect the flow at all. Right: development of velocity in the simulation. The y-axis shows the velocity magnitude, and the x-axis represents the position of the line.

This simulates a default 2D flow in a blood vessel without an eruption. The permeability is set to $1.36 \cdot 10^{-9} \mu m^2$. With some assumptions, this can be represented as blood flow in a healthy zebrafish. The line shown in fig. 5.3.5 is set to go straight through the area where an eruption is expected. This line will be used in the other simulations to analyze the velocity development.

Case 2: Increased Permeability

In this case, the permeability is increased, such that we can study the beginning of an eruption. We are simulating the start of an eruption and a few steps after the eruption has occurred. Remember that we use a gap that has an opening in the viscous domain equal to $2.8 \mu m$ and expands to a $4 \mu m$.

We assume that this is related to the case where the blood cells start to leak through the walls, and the eruptions start to build up. The simulations can be compared to the eruption that is illustrated in fig. 2.4.13), in the time frame of 30 to 50 seconds. A magnitude profile with for different permeability is illustrated as following:

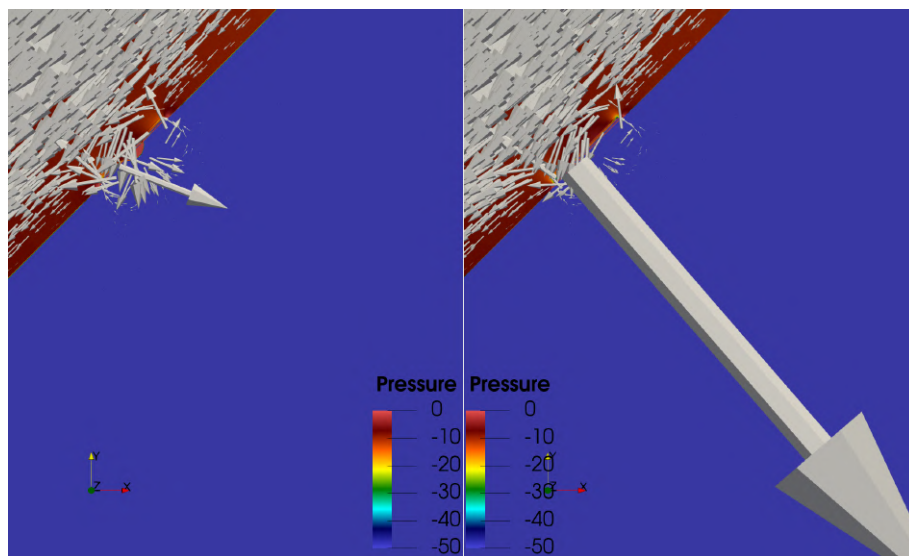


Figure 5.3.6: A zoomed-in view of the simulations for the coupled flow in an enclosed mesh. Left: simulation with permeability: $1.36 \cdot 10^{-5} \mu m^2$, right: simulation with permeability: $1.36 \cdot 10^{-4} \mu m^2$.

This simulates a specific outburst with the variables reviewed in table 5.1 and section 5.3.2. From the left image in fig. 5.3.6, the gap has opened up, and the eruption starts. The closed mesh causes the pressure to build up.

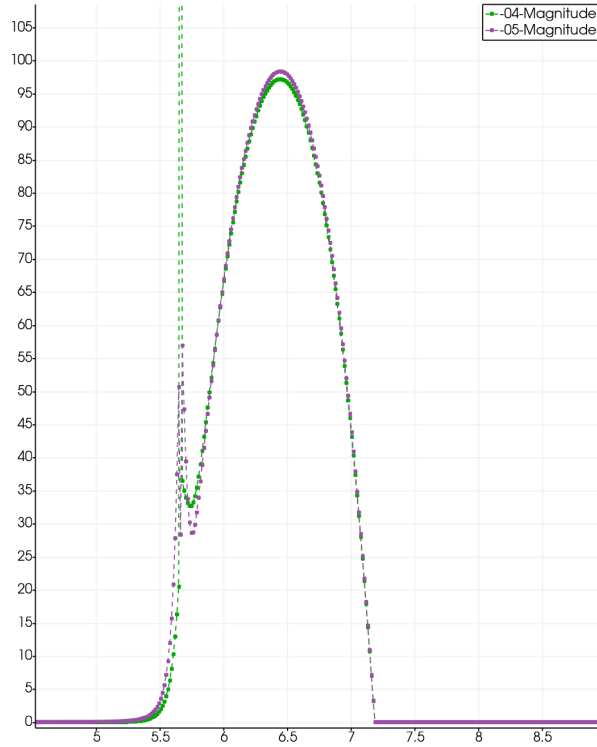


Figure 5.3.7: A zoomed-in view of the line plot, with the same line as in fig. 5.3.5.

The profile for permeability equal to $1.36 \cdot 10^{-5} \mu m^2$ and $1.36 \cdot 10^{-4} \mu m^2$ are represented with the purple and green line in fig. 5.3.7. We can see that there is some development around the gap. The maximum values calculated with the `Plot over line` function in ParaView for the velocity magnitude at the gap, are around $56 \frac{\mu}{s}$ and $360 \frac{\mu}{s}$ for permeability $1.36 \cdot 10^{-5} \mu m^2$ and $1.36 \cdot 10^{-4} \mu m^2$.

Case 3: Hyperpermeability

The last case that we are going to study is the hyperpermeability, which is equivalent to a fully developed eruption. The permeabilities are between $1.36 \cdot 10^{-3} \mu m^2$ and $1.36 \cdot 10^{-2} \mu m^2$. Increasing the permeability more than this will not affect the velocity at the gap. The simulations for permeability higher than this are shown in appendix A. With these permeabilities, we can assume this is how the eruption acts in the last timestamp of the burst. The simulation can be compared to fig. 2.4.13, around 60 to 80 seconds after burst.

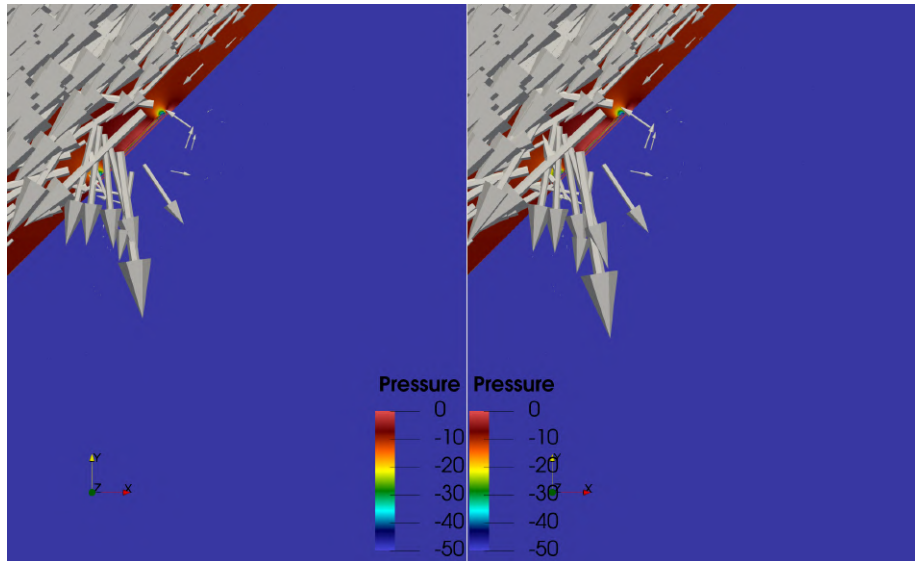


Figure 5.3.8: Simulation of the coupled flow in an enclosed mesh. Left: permeability $1.36 \cdot 10^{-3} \mu m^2$, right: permeability $1.36 \cdot 10^{-2} \mu m^2$.

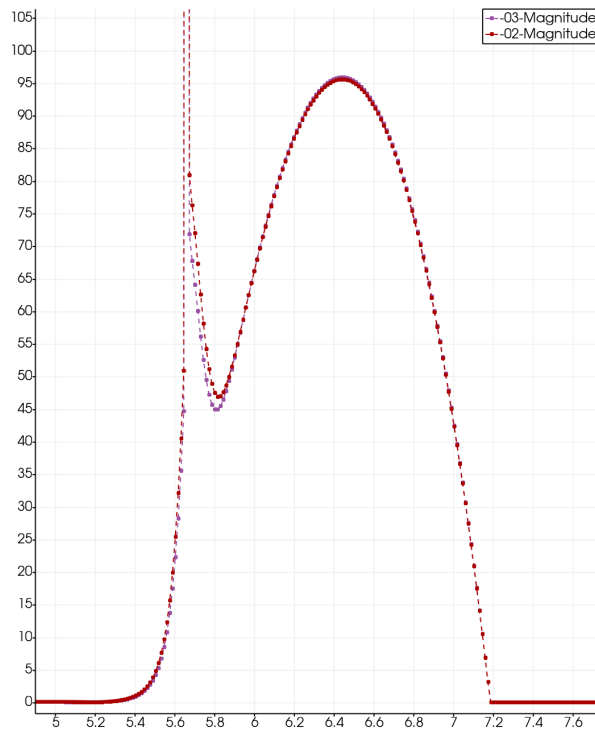


Figure 5.3.9: A zoomed-in view of the line plot, with the same line as in fig. 5.3.5.

The maximum value of the velocity magnitude at the gaps are $810 \frac{\mu}{s}$ and $925 \frac{\mu}{s}$ for permeability set to $1.36 \cdot 10^{-3} \mu m^2$ and $1.36 \cdot 10^{-2} \mu m^2$. There are minimal

changes in the velocity for in the simulations with permeability equal to $1.36 \cdot 10^{-3} \mu m^2$ and $1.36 \cdot 10^{-2} \mu m^2$, shown in fig. 5.3.8. We will assume that the flow has started to stabilize at this point. Hyperpermeability is achieved.

5.3.3 Simulation: Eruption in a Open Grid

Now that we have studied a general eruption within a closed grid, the next step is to look at an open grid. We will assume that the leak can stream further than the red rectangle shown in fig. 3.3.3. In section 3.2, we concluded that the fluid flow between the cell-to-cell junction pursues the same principals as a free viscous flow. We will rewrite our code such that stokes flow is implemented in the junction in the porous domain.

Preliminaries

The permeability is evenly distributed between $1.36 \cdot 10^{-9} \mu m^2$ and $1.36 \cdot 10^{-2} \mu m^2$. There is a significant change to the Dirichlet conditions, such that the streamlines can flow more freely out of the grid. The only boundary set for this case is, Stokes flow at v_{in} . The no-slip conditions are removed from the boundaries in the porous domain. These conditions will produce the same manufactured solution as in eqs. (5.1) to (5.3).

There is limited research on the gap size of an eruption inside an infected zebrafish. However, there have been quite a few experiments for the same phenomena in mice. In the paper by Stefan Wilhelm et al. [53], they conclude that the endothelial cell contractions are around $100nm - 500nm$.

The mesh has been reproduced with a gap set to $300nm$, which expands to $1400nm$ at the porous end.

Simulations

The assumptions done for this simulation gives a drastic change to the listing 4.2. The p_domain in the bilinear form will be changed, to Stokes flow. The no-slip Dirichlet boundaries will be removed, and we will get the following boundary conditions:

```
inflow = Expression(("0", "180*(6-x[0])*(7.5-x[0])"), degree=2)
bc3_v = DirichletBC(W.sub(0), inflow, boundaries_mf, mark["v_inflow"])
bcs = [bc3_v]
```

The method of manufactured solutions is the same as for the previous experiment 5.3.2, and we will continue to work on a two-dimensional mesh.

Case 1: Low Permeability

A simulation of the resting case in an open grid, with the permeability set to $1.36 \cdot 10^{-9} \mu m^2$, are shown in the figure below.

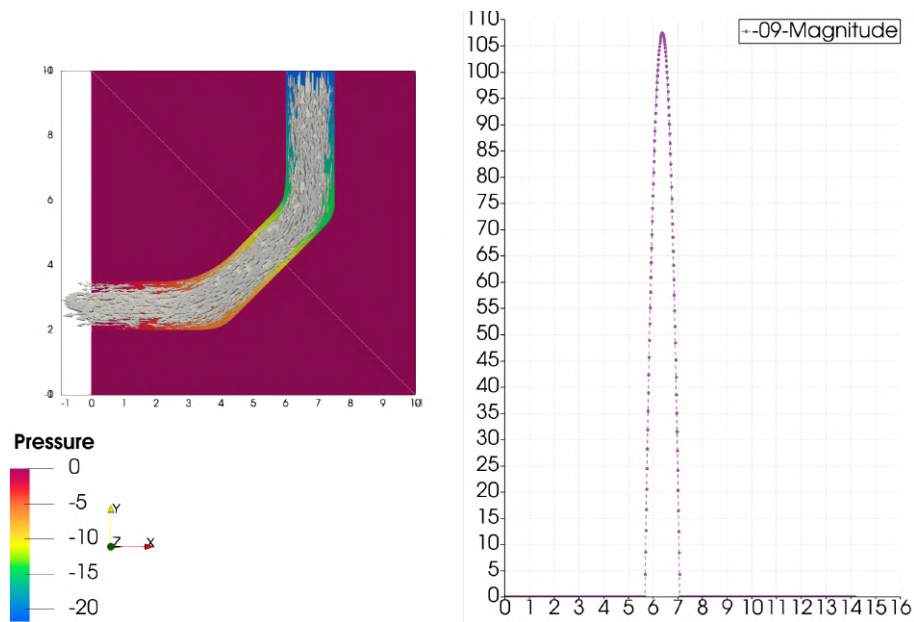


Figure 5.3.10: Left: Simulation of viscous and porous flow in an open mesh with low permeability. In this case, the gap does not affect the flow at all. Right: development of velocity in the simulation. The y-axis shows the velocity magnitude, and the x-axis represents the position of the line.

This gives a simulation of a steady flow with no eruption. The choice of permeability leads to a closed gap.

Case 2: Increased Permeability

Simulations of increased permeability, where the gap starts to open up slowly.

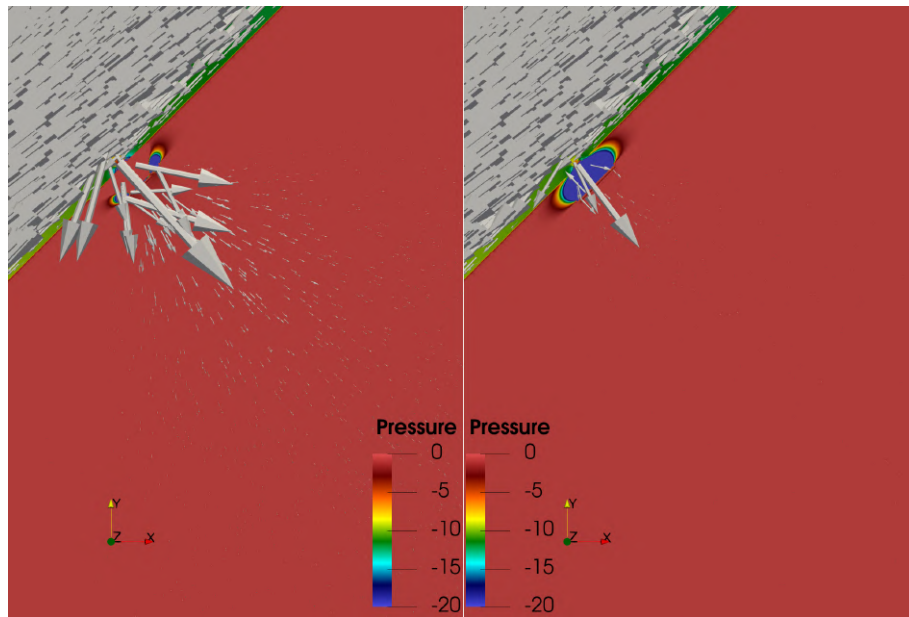


Figure 5.3.11: A zoomed-in view of the simulations for the coupled flow in an open mesh. Left: simulation with permeability: $1.36 \cdot 10^{-5}$, right: simulation with permeability: $1.36 \cdot 10^{-4}$.

In the left image in fig. 5.3.11, the gap has opened up, and the flow starts to leak into the porous media.

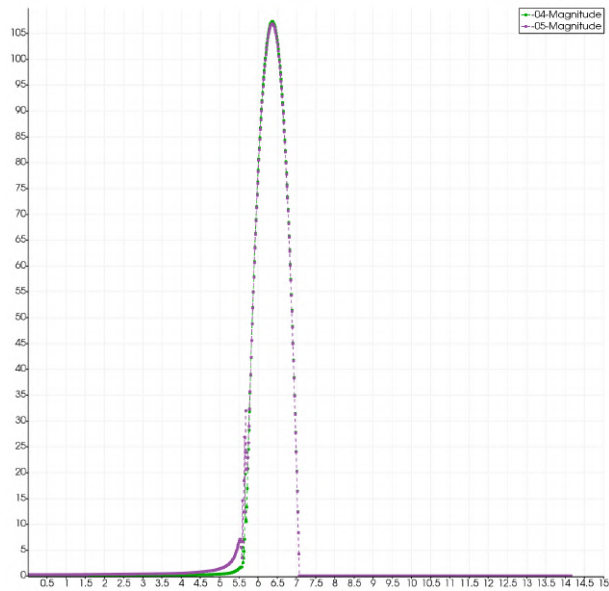


Figure 5.3.12: Plot over line for the simulations in fig. 5.3.11, with the line at the same location as in fig. 5.3.10.

The profile for permeability equal to $1.36 \cdot 10^{-5} \mu m^2$ and $1.36 \cdot 10^{-4} \mu m^2$ are represented with purple and green lines. From fig. 5.3.12, we observe that the **Plot over line** shows an increase of velocity vectors further away from the gap.

We will look closer into the velocity development of the eruption in section 5.4.1.

Case 3: Hyperpermeability

The last simulations show the behavior of the leakage close to a stabilized eruption. This occurs when hyperpermeability is archived.

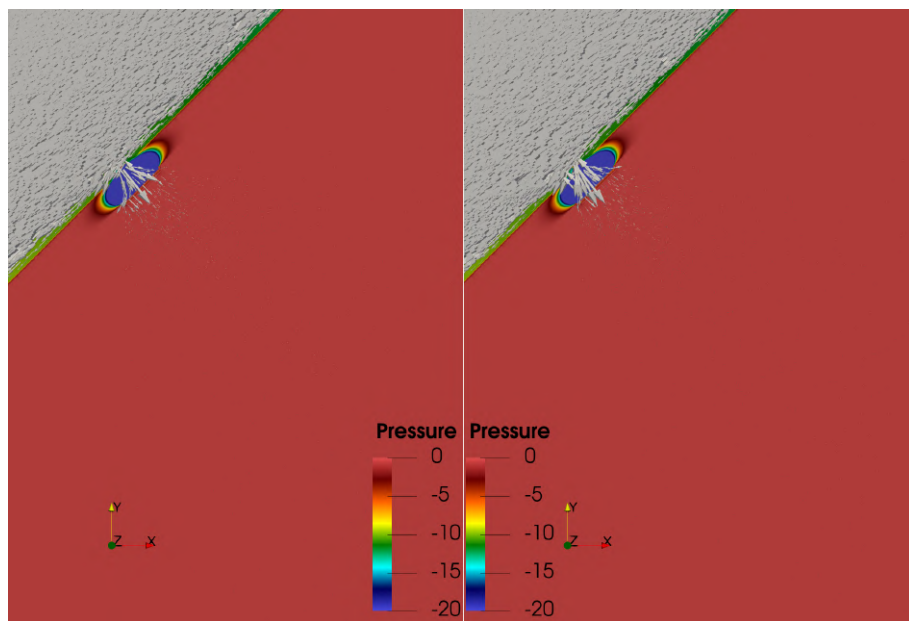


Figure 5.3.13: Simulation of the coupled flow in an open mesh. Left: permeability $1.36 \cdot 10^{-3} \mu m^2$, right: permeability $1.36 \cdot 10^{-2} \mu m^2$.

We assume that the simulation to the right in fig. 5.3.13, illustrate a fully evolved eruption. Increasing the permeability will not affect the eruption.

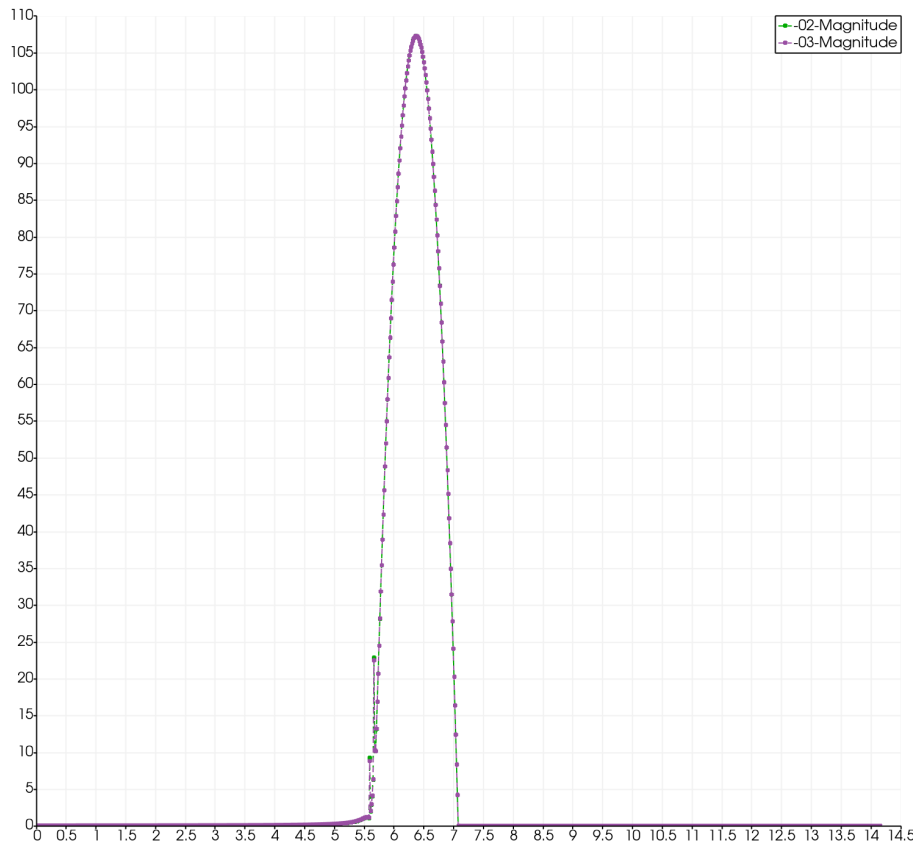


Figure 5.3.14: plot over line for the simulations in Figure 5.3.13.

From fig. 5.3.14, we observe that there are minimal changes to the velocity development between permeability $1.36 \cdot 10^{-3} \mu m^2$ and $1.36 \cdot 10^{-2} \mu m^2$.

5.4 Results

In this section, we will compare the stationary results from the simulations with the non-stationary observations provided by Vollestad [57]. The method described in section 4.4.1 will be used for the comparisons.

5.4.1 Velocity Development in the Coupled Scheme

The inflow has been chosen, so the velocity in the viscous domain is comparable with the results from section 2.2.3 in chapter 2. By increasing the permeability for both scenarios, we get the following velocity:

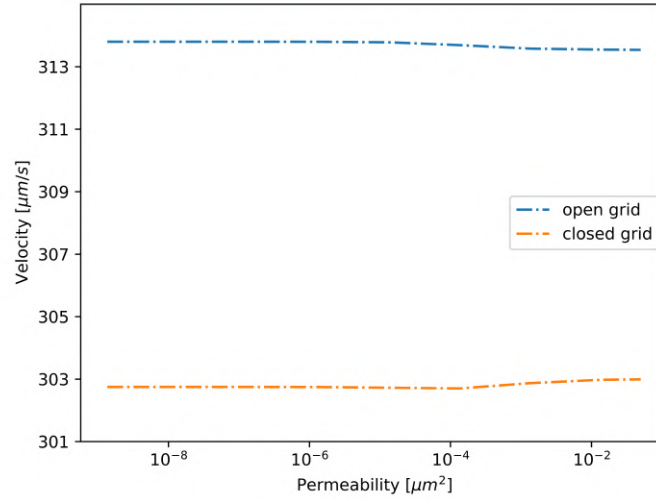


Figure 5.4.15: Velocity development inside the simulated blood vessel with increasing permeability. Note that there are small changes to the values on the y-axis.

The mean velocity equals to $314 \frac{\mu m}{s}$ for an open grid and $303 \frac{\mu m}{s}$ a closed grid. The values agree with the velocity calculated in section 2.2.3.

Since we assume that the gap closes after it reaches hyperpermeability, indicates that this is the last step of the development. The eruption is fully developed at this point. The top image in fig. 5.4.16 shows the velocity development in the porous and interface domain. The values have been calculated from the mathematical expression for velocity in section 4.4.1 for each experiment with permeability. We will study the development in the closed grid first. The velocity in the simulated cases provides a representation of how the development is in the real-life case. However, the velocity development does not match the results provided by Vollestad [57]. The difference in gap sizes causes the mismatch. The velocity has a much bigger amplitude than the calculated velocity.

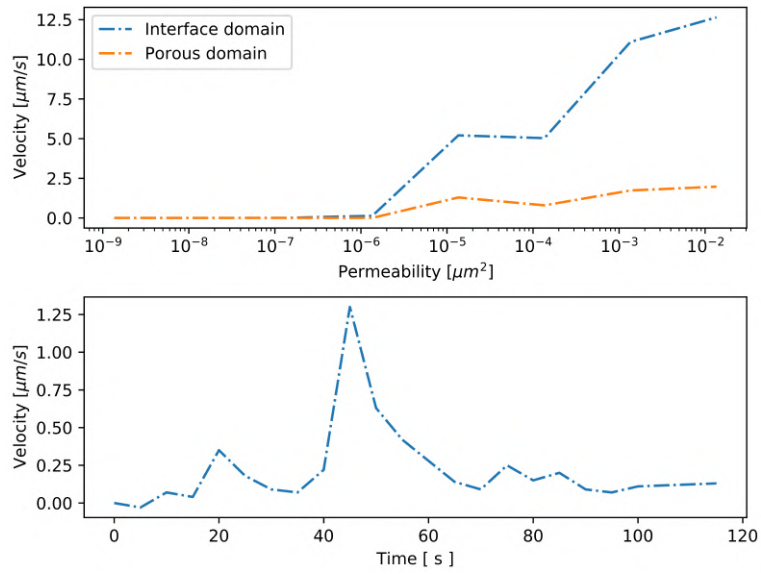


Figure 5.4.16: Top: Velocity development in porous and interface domain, when permeability increases. Bottom: calculated velocity development over time with laminar eruption from the paper by [57].

The next step is to look at the velocity development for an open grid, which represents the case where the flow can stream out of the $100 \times 100 \mu\text{m}$ area. The same velocity developments will be studied for this case. The development of velocity throughout increased permeability in the interface and the porous domain is shown in fig. 5.4.17. The velocity in the simulated cases provides an excellent representation of the development in a real-life case.

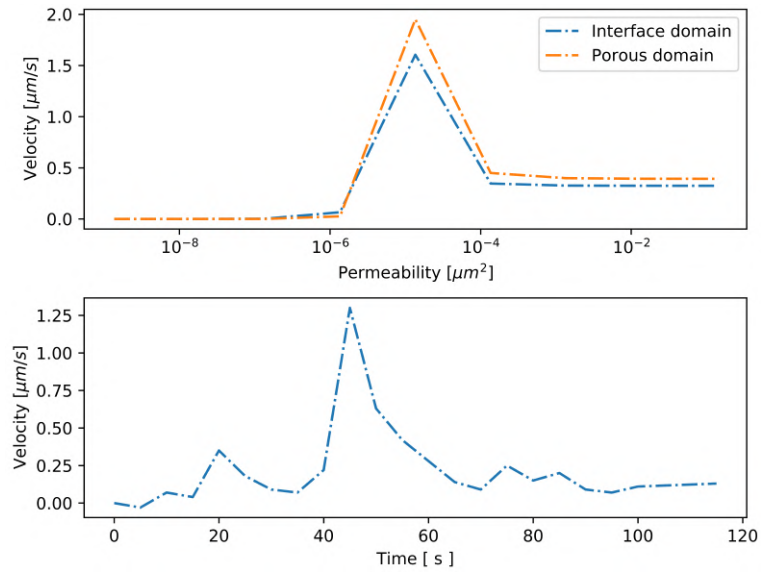


Figure 5.4.17: Top: Velocity development in porous and interface domain, when permeability increases, for an open grid with a gap size of $300nm$. Bottom: calculated velocity development over time with laminar eruption from the paper by [57].

The amplitude for the simulated case is closer to the calculated case provided by Vollestad [57], compared to the closed grid.

5.4.2 Pressure Development in the Coupled Scheme

We have calculated the analytical solution to a laminar case, shown in eq. (2.7), where we show the pressure change for the entire eruption. The calculated value will be used for comparison with the numerical results that we have collected.

The mathematical expression, shown in section 4.4.1, is used to calculate the pressure for each simulation. This is plotted against the increased permeability in. The pressure change in the viscous domain for both cases are given by:

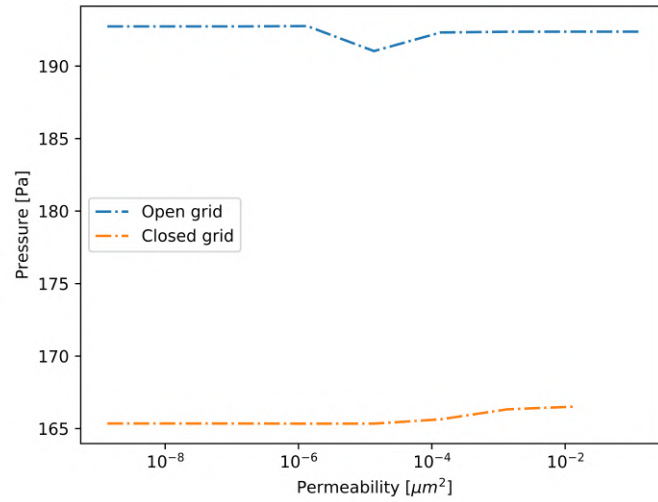


Figure 5.4.18: Pressure development against increased permeability in the viscous domain. Note that the changes in the y-axis are small.

We will separate the results for pressure in the other domains since the calculated ratio between a closed and open grids are significant. Since we have a closed grid, the pressure will accumulate.

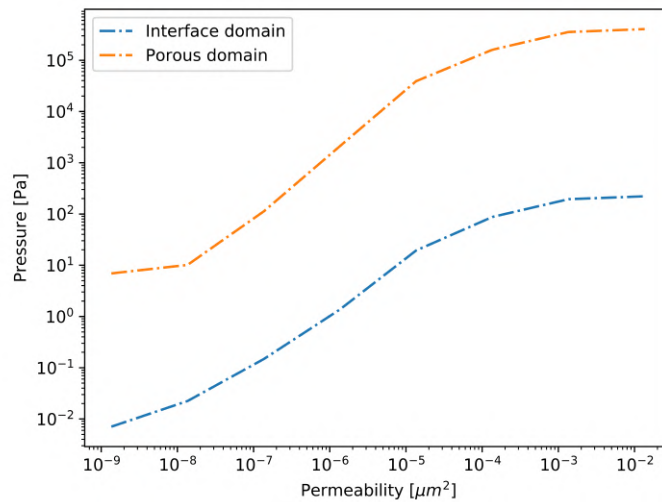


Figure 5.4.19: Pressure development against increased permeability in the interface and porous domain for closed grid. The changes in the y-axis are represented logarithmically.

The pressure development generated in the closed grid is unrealistically large.

An open grid results in a better representation of the pressure development.

The results in the porous and interface domain are shown separately.

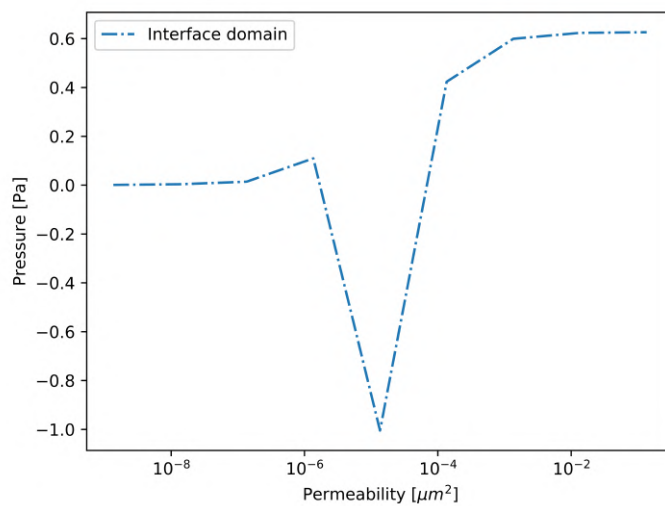


Figure 5.4.20: Pressure development against increased permeability in the interface domain for an open grid.

Since the interface domain is located between the viscous and porous domains, the flow passing through the interface domain result in a negative pressure difference.

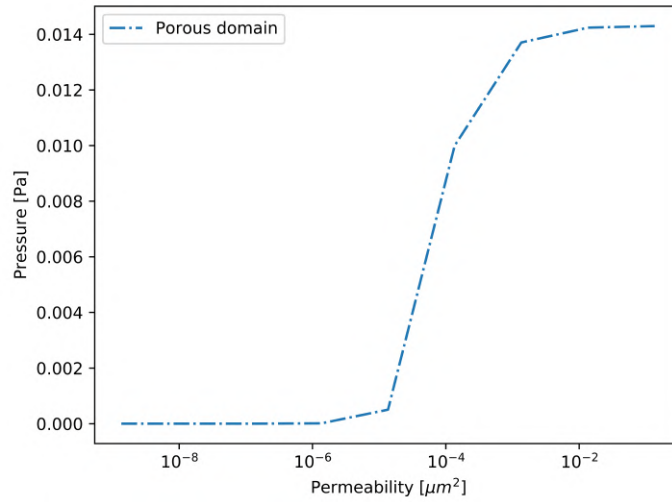


Figure 5.4.21: Pressure development against increased permeability in the porous domain for an open grid. Note that the changes in the y-axis are small.

The comparison between simulated and calculated pressure changes are reviewed in the discussion.

Chapter 6

Summary and Discussion

In this thesis, we have simulated a model for the evolution of a burst in a blood vessel covered by tissues, using a unified mixed discretization of the coupled Darcy–Stokes problem. The goal was to simulate a stationary eruption with increased permeability, and compare it to the theoretical assumption done in section 2.3, with the burst described in section 2.4.1.

In this chapter, we will summarize and discuss how each part has effected our results and describe how this can be improved in further work.

6.1 Summary

6.1.1 The Coupled Darcy-Stokes Equation

The coupled Darcy-Stokes scheme is generated with a unified strategy, represented weakly with L^2 -formulation, which was discretized with the Taylor-hood elements. Implementing this method is straightforward in FEniCS software and has a low computationally expensive when we work on a two-dimensional space with the right choice of mesh. This has made possible to test many different cases and adjust the Dirichlet and Neumann conditions to generate the simulations. However, Taylor-Hood elements are unable to reproduce the physical pressure discontinuity at the interface between the viscous and porous domains.

The numerical and mathematical techniques used in this thesis are comparable with the research done by Karper et al. [21], Abouorm et al. [1], and Ellingsrud [11]. The papers solve the coupled Darcy-Stokes problem, with the same L^2 -formulation as in this thesis. The Taylor-Hood element discretization is also introduced in Ellingsrud [11], Karper et al. [21].

6.1.2 The Domain

Each blood vessel has a unique geometry, surrounded by different shapes of cells. Throughout the thesis, we have held to the same geometry, where we show the simulations in a $100 \times 100 \mu m$ area. The most notable changes done to the domain are testing with different gap sizes and boundary conditions, such that we get a clear view of how these changes can influence the problem.

We have done simulation for two different gap sizes, with roughly the same numerical method. The gaps have been generated such that they have an oblique growth outwards, which ensures that the fluid gets a smooth transition from the viscous domain and out to the porous domain. The location of the gaps has been chosen carefully around the center of the bend in the vessel. The large-scaled gap in section 5.3.2 gives a clear view of how the flow behaves around the gap. Another experiment with a large-scaled gap with an open grid is shown appendix A. However, this is an unrealistic choice of gap size. The smaller gap size in section 5.3.3 is chosen based on the research in the paper by Stefan Wilhelm et al. [53]. Implementing those circumstances will generate simulations which can be related to real case scenarios discussed in chapter 2.

6.1.3 Boundaries

In section 2.2.1, we looked at different types of observed eruptions in a stationary case, emerged from some arbitrary part in the zebrafish body. These illustrations show different types of eruptions. Simulating a precise case is a difficult task, considering that each eruption occurs uniquely. However, introducing the right preliminaries results in useful simulations.

The preliminaries shown in section 5.3.2 and 5.3.3 for the boundaries have been chosen carefully. They show different layouts for the surrounding tissues. The boundary conditions for simulations shown in section 5.3.2 illustrates a blood vessel in an enclosed surrounding. This prevents the leakage from expanding any further. While in section 5.3.3, we simulated the eruption for an open grid. However, the conditions are set such that the leakage can continue steaming in the cell-to-cell junctions in the porous domain. The conditions introduced for an open grid in section 5.3.3 generated good simulation for comparison with the calculations and studies shown in section 2.4.1.

The simulations executed in chapter 5 only show the development in a $100 \times 100 \mu m$ area, which makes it challenging to study leakages further than the grid.

6.1.4 Simulations

The first numerical experiment (section 5.3.2) shows a simulation of an eruption into a tissue using a closed grid. This shows how the eruption behaves in a compact location in the circulation, with tight tissue structure.

In the textbook by Molnar and Gair [34] the relationships between velocity and

pressure in the circulations have been studied. By using the results provided by the paper and the results from section 5.3, we can relate the grid used in the numerical experiments to a suitable vessel. To do this, we need to assume that the numerical experiments agree with the physiology of the tissue and blood vessels in a zebrafish. In the closed grid, the changes in pressure and velocity are large. From the textbook by Molnar and Gair [34] we know that this behavior occurs in the transition from the artery to capillary close to the heart. The case with an open grid gives us a significantly smaller pressure change, while the velocity sustained the same as in the closed grid. We can therefore assume that this grid can be located in the veins.

From the research by Stefan Wilhelm et al. [53], we know that the gap size should be between 100 and 500. By using a gap size equal to 300, we see that the numerical experiments match with the calculations (shown in section 6.1.5).

The simulation has been done for permeabilities between $1.36 \cdot 10^{-9} \mu m^2$ and $1.36 \cdot 10^{-2} \mu m^2$. For each experiment, the permeability is multiplied by ten, until we achieve a stable velocity and pressure. From the simulations, we see that the eruptions stabilizes around permeability $1.36 \cdot 10^{-2} \mu m^2$. This case represents a fully evolved eruption. Simulations for permeability higher than this are added in appendix A. These simulations confirm that there are no significant changes to the eruption when increasing the permeability further than $1.36 \cdot 10^{-2} \mu m^2$. Hence our simulations illustrate the evolution of an eruption from the beginning to a stable state. This can be compared with the experiment shown in section 2.4.2, where the laminar flow was fully evolved after 80 seconds, before it stabilizes.

As mentioned in [16, 32, 55], the cause of endothelial cell contractions, is increased permeability. The results from the stationary experiments with increased permeability supports the hypothesis introduced in these reports.

6.1.5 Velocity and Pressure Development

In chapter 5, numerical calculations for velocity and pressure development in a formed eruption was accomplished. Using the results from the experiment in section 5.3.3 and the calculation in section 5.4.1, we can determine a reasonable relation between the velocity development in time and the numerical experiments with increased permeability. The evolution of an eruption over time are illustrated thoroughly in section 2.4.1, and in the paper by Vollestad [57]. With these correlations, the stationary development of velocity in an open grid can be compared with the eruption emerging over time. This is shown in table 6.1.

Table 6.1: Comparison between the increased permeability used in the simulations of an eruption, shown in section 5.3.3, and the time-evolving eruption represented in section 2.4.2.

Permeability	Time
$1.36 \cdot 10^{-9} \mu m^2$	0 s
$1.36 \cdot 10^{-6} \mu m^2$	40 s
$1.36 \cdot 10^{-5} \mu m^2$	50 s
$1.36 \cdot 10^{-4} \mu m^2$	60 s
$1.36 \cdot 10^{-3} \mu m^2$	80 s
$1.36 \cdot 10^{-2} \mu m^2$	100 s
$1.36 \cdot 10^{-1} \mu m^2$	120 s

We used the same approach to analyze the pressure development. The results submitted in section 5.4.2 shows the pressure development for increased permeability in the numerical experiments. The pressure difference obtained by the numerical experiment in the open grid (in the porous domain) is 0.014 Pa , shown in fig. 5.4.21. This value is relatively close to the calculations done for a pressure difference in an eruption in a physical experiment, which results in 0.03340 Pa , shown in eq. (2.7). This value has been governed from the experiments performed by Vollestad [57]. Furthermore, we can conclude that the numerical calculations executed for the pressure difference and velocity is a reasonable estimate.

6.2 Limitations

Because of limited memory resources, the generated mesh size resulted in high deformation near the gap. This yields particularly for the simulations shown in fig. 5.3.6. ParaView was used for visualizing the velocity and pressure. However, The resulting images obtained from the numerical experiment are large (several GB), and manipulating with such data is difficult on a standard computer with 8 or 16 GB of RAM. Hence memory issues prevent us from analyzing the data with specific techniques that are provided in ParaView. Furthermore, using FEniCS on grids with a large number of elements resulted in memory issues. The simulations for permeability higher than $1.36 \cdot 10^{-2} \mu m^2$, in some cases, were not possible to obtain. There is some disadvantage when Taylor-Hood finite elements are implemented. There is no element-wise mass conservation for the continuous pressure space. Taylor-hood elements are not pointwise divergence-free.

6.3 Further Work

The goal in this thesis was to simulate the simultaneous flow in both viscous and porous flow with increased permeability for the blood vessel wall (endothelial cells), which resulted in a model for the development in an eruption. However, we only simulated the experiment using a simple two-dimensional geometry.

In this experiment, the permeability is increased for each experiment with a factor of 10. Each simulations show the results in an equilibrium stage. However, increasing permeability with a smaller factor would give a detailed view of the eruption in a much smaller stage, or even converting over to a non-stationary

problem. The next step is to use the numerical scheme to solve the problem on a more realistic three-dimensional geometry. Moreover, extending the conditions in the Stokes equation to the Navier-Stokes equations, such that we can apply the Navier-Stokes-Darcy model [9].

6.4 Conclusion

In this thesis, a stationary L^2 -formulation of coupled Darcy-Stokes discretized with Taylor-Hood elements was implemented on a two-dimensional mesh, illustrating a blood vessel and its surrounding. This resulted in a simplified model that can be used to represent complicated natural and medical phenomena.

One of the most critical values in this thesis was permeability, which has an essential impact on eruption development. The numerical method in this thesis presents the possibility to study the growth of a burst throughout increasing the permeability. Furthermore, analyzing the velocity and pressure of an eruption in stationary simulations generated a robust model of how the fluid behaves.

In the simulations and during the result-analysis, we noticed that our numerical methods introduce an excellent method for visualization of an eruption, which occurs from arbitrary contractions in endothelial cells. A relatively good comparison with the calculations done by Vollestad [56, 57] has been accomplished. Furthermore, a good understanding of what the generated simulation has in common with the experiments provided by Resseguier [41, 42, 43, 44].

Appendix

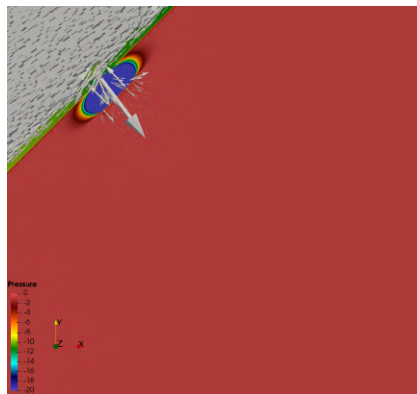
Appendix A

Additional simulations

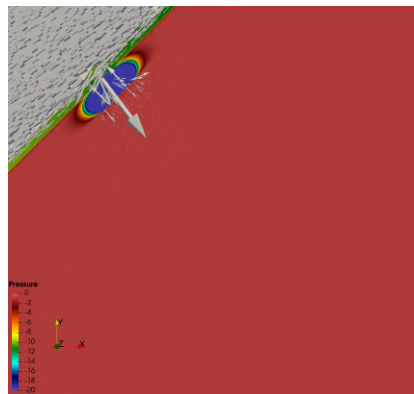
In this appendix, additional simulations are introduced. When it comes to visualizing of the simulations, the same properties as in section 5.3.2 and Section 5.3.3 have been used.

High permeability simulations for an open grid and $300nm$ gap size

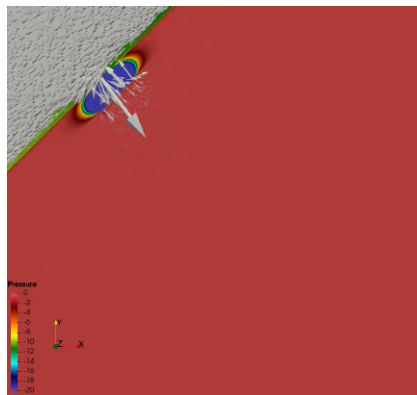
The figure below shows the simulations for the stabilized eruption in section 5.3.3, which clarifies that increasing permeability does not affect the eruption.



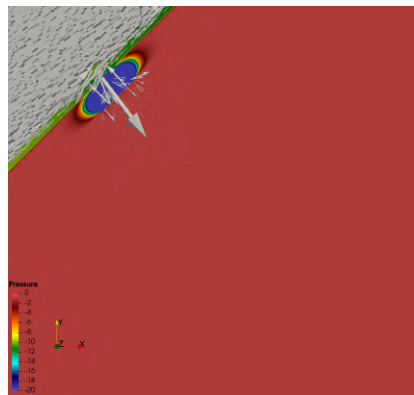
(a) Permeability: $1.36 \cdot 10^{-1} \mu\text{m}^2$



(b) Permeability: $1.36 \cdot 10^0 \mu\text{m}^2$



(c) Permeability: $1.36 \cdot 10^1 \mu\text{m}^2$



(d) Permeability: $1.36 \cdot 10^2 \mu\text{m}^2$

Figure 1.0.1: The stabilized simulations of an eruption in an open grid.

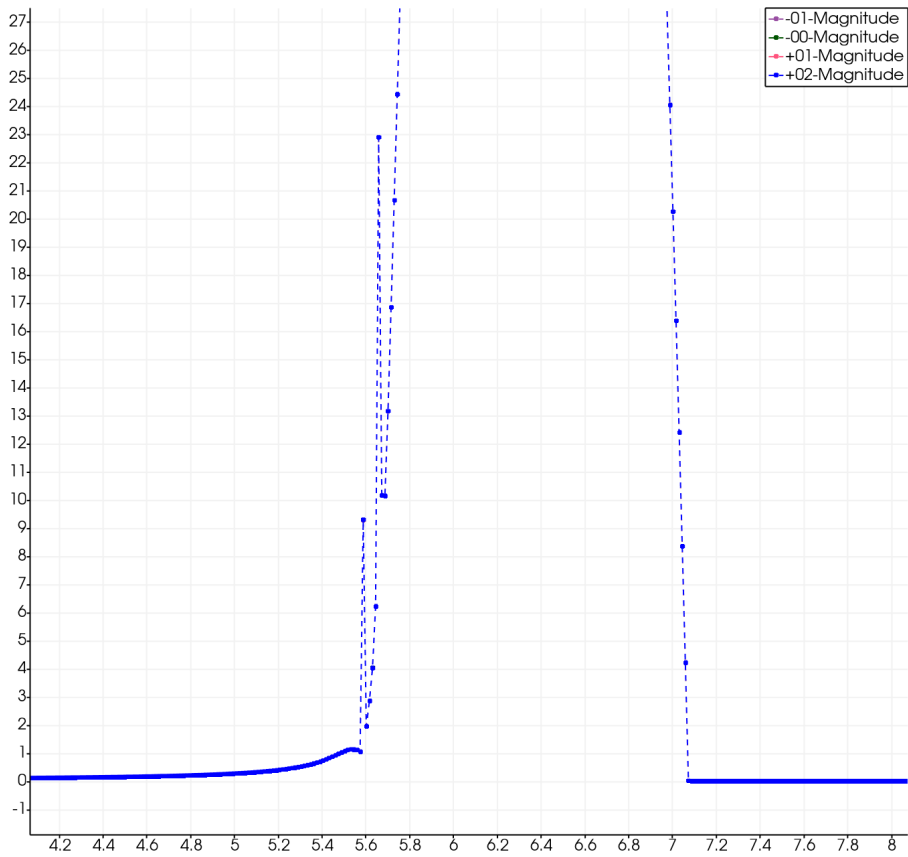
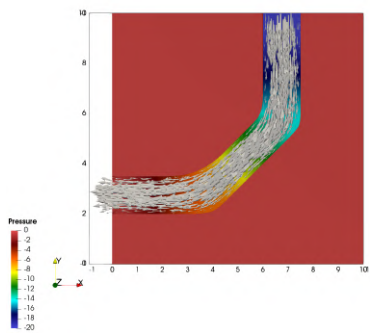


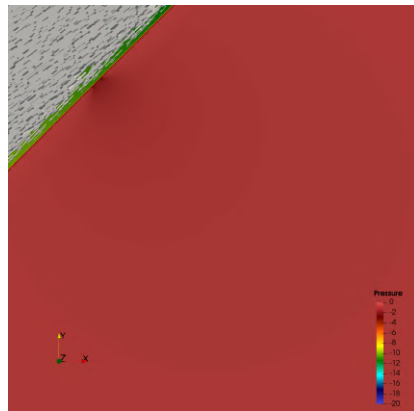
Figure 1.0.2: Plot over line in for permeabilities: $1.36 \cdot 10^{-1} \mu m^2$ - $1.36 \cdot 10^2 \mu m^2$ in an open grid and $300nm$ gap size. All the values overlap; hence there is no change to the velocity. Furthermore, there are no changes to the pressure.

Simulations for a closed grid and $300nm$ gap size

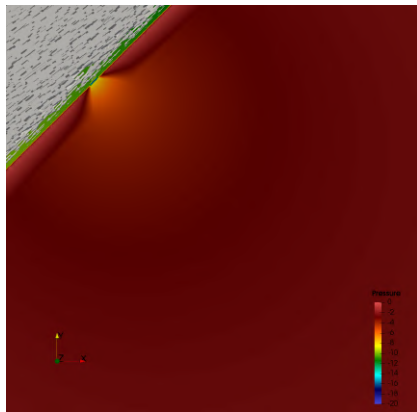
The presented simulations have the same eruption gap as the simulations shown in section 5.3.3, with the boundary condition in section 5.3.2. The simulations are shown below.



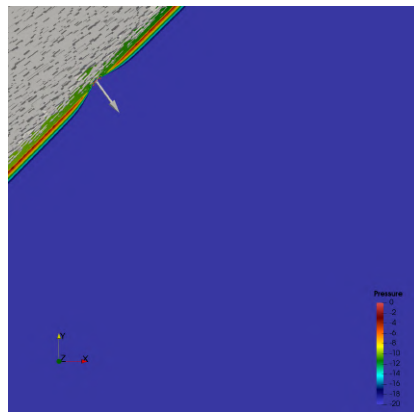
(a) Permeability: $1.36 \cdot 10^{-9} \mu m^2$



(b) Permeability: $1.36 \cdot 10^{-8} \mu m^2$



(c) Permeability: $1.36 \cdot 10^{-7} \mu m^2$



(d) Permeability: $1.36 \cdot 10^{-6} \mu m^2$

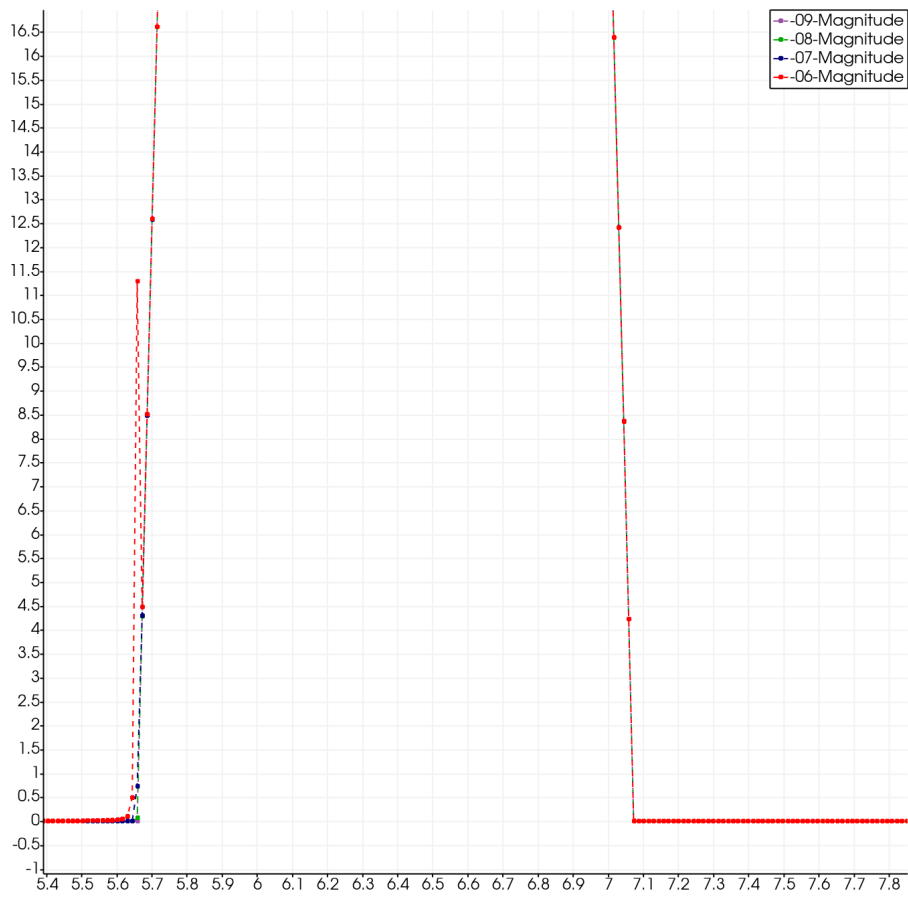
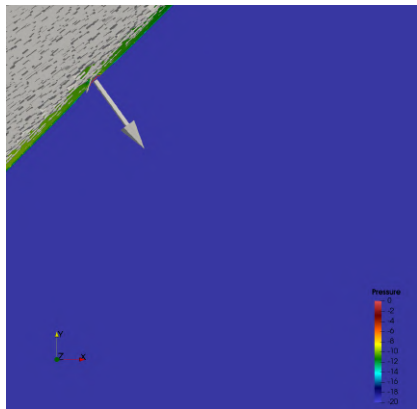
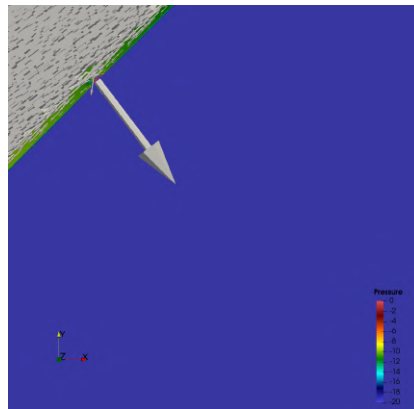


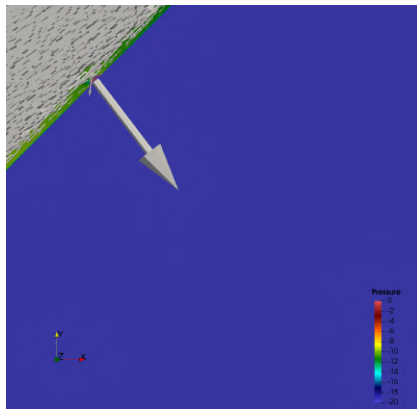
Figure 1.0.4: Plot over line for permeability: $1.36 \cdot 10^{-9} \mu m^2$ - $1.36 \cdot 10^{-6} \mu m^2$ in an closed grid and $300nm$ gap size.



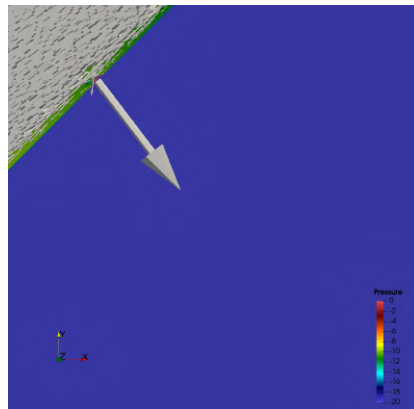
(a) Permeability: $1.36 \cdot 10^{-5} \mu\text{m}^2$



(b) Permeability: $1.36 \cdot 10^{-4} \mu\text{m}^2$



(c) Permeability: $1.36 \cdot 10^{-3} \mu\text{m}^2$



(d) Permeability: $1.36 \cdot 10^{-2} \mu\text{m}^2$

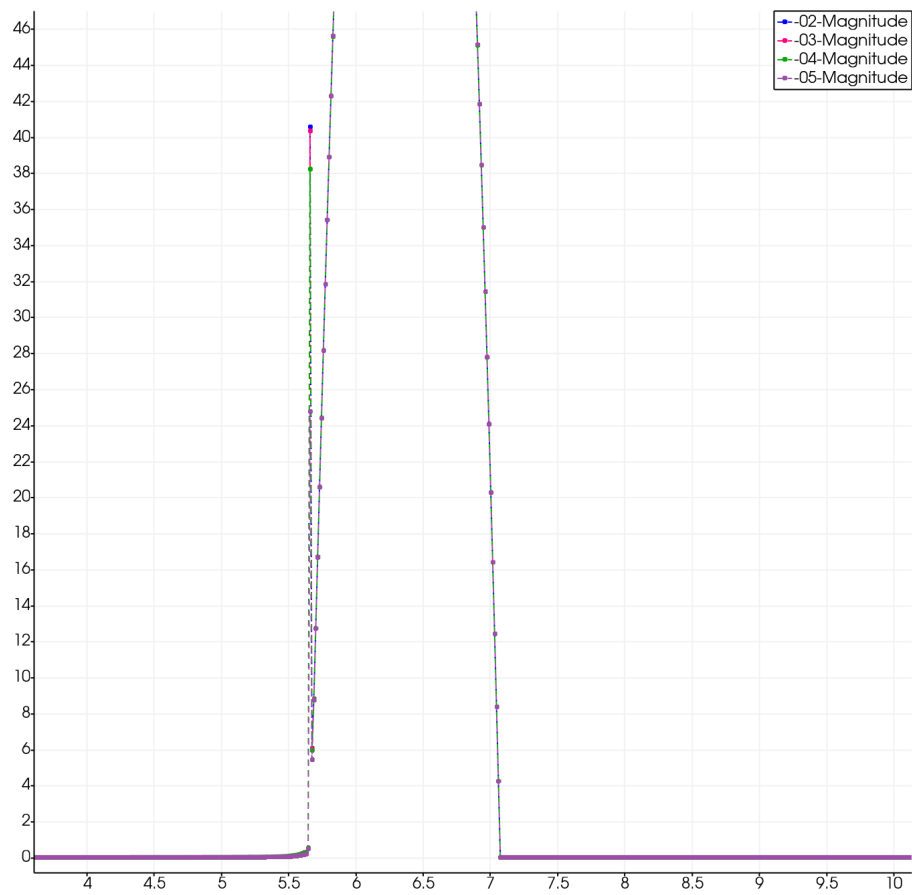


Figure 1.0.6: Plot over line for permeability: $1.36 \cdot 10^{-5} \mu m^2$ - $1.36 \cdot 10^{-2} \mu m^2$ in an closed grid and $300nm$ gap size.

Velocity and Pressure Development

The following figures illustrate the velocity and pressure development in a closed grid and 300nm gap size, for each subdomain.

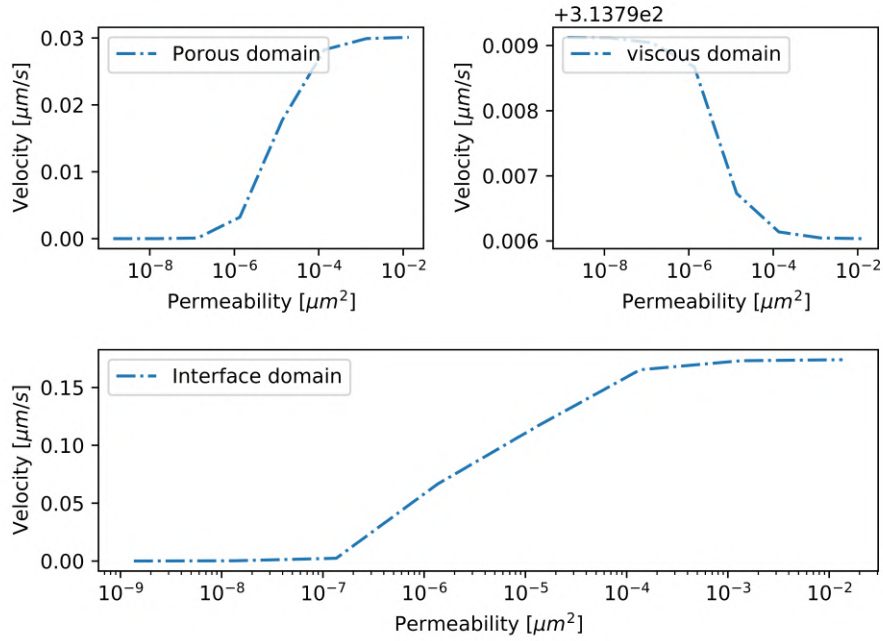


Figure 1.0.7: The velocity development for each subdomain in a closed grid and 300nm gap size. Note the small changes in the y-axis.

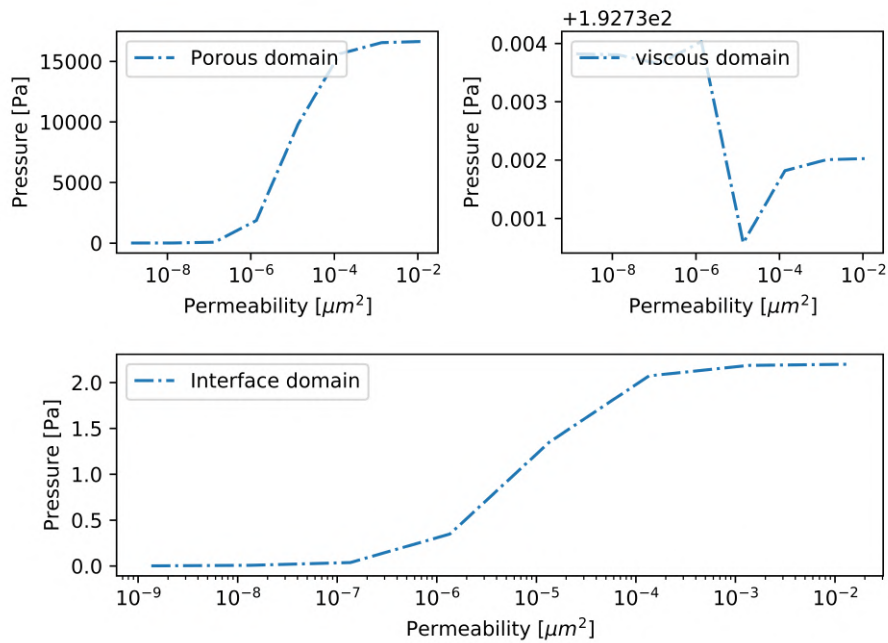
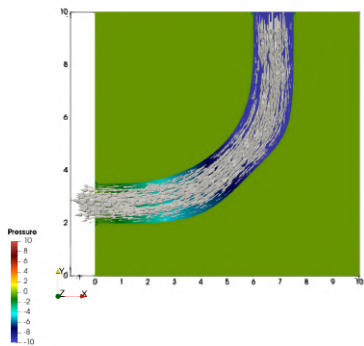


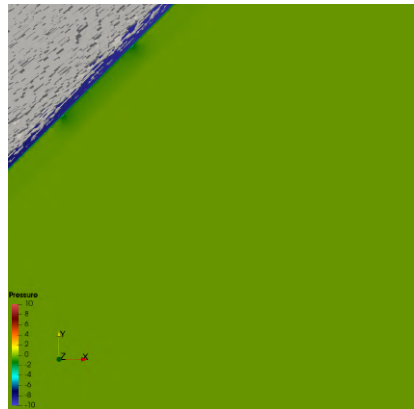
Figure 1.0.8: The pressure development for each subdomain in a closed grid and $300nm$ gap size. Note the small changes in the y-axis.

Simulations for a open grid and $4\mu m$ gap size

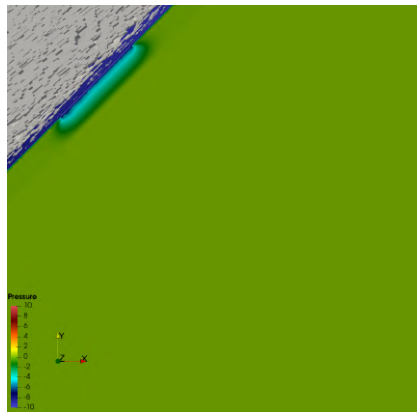
The presented simulations have the same eruption gap as the simulations shown in section 5.3.2 with the boundary condition in section 5.3.3. The simulations are shown below.



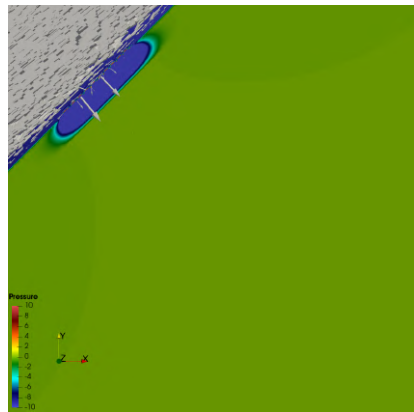
(a) Permeability: $1.36 \cdot 10^{-9} \mu m^2$



(b) Permeability: $1.36 \cdot 10^{-8} \mu m^2$



(c) Permeability: $1.36 \cdot 10^{-7} \mu m^2$



(d) Permeability: $1.36 \cdot 10^{-6} \mu m^2$

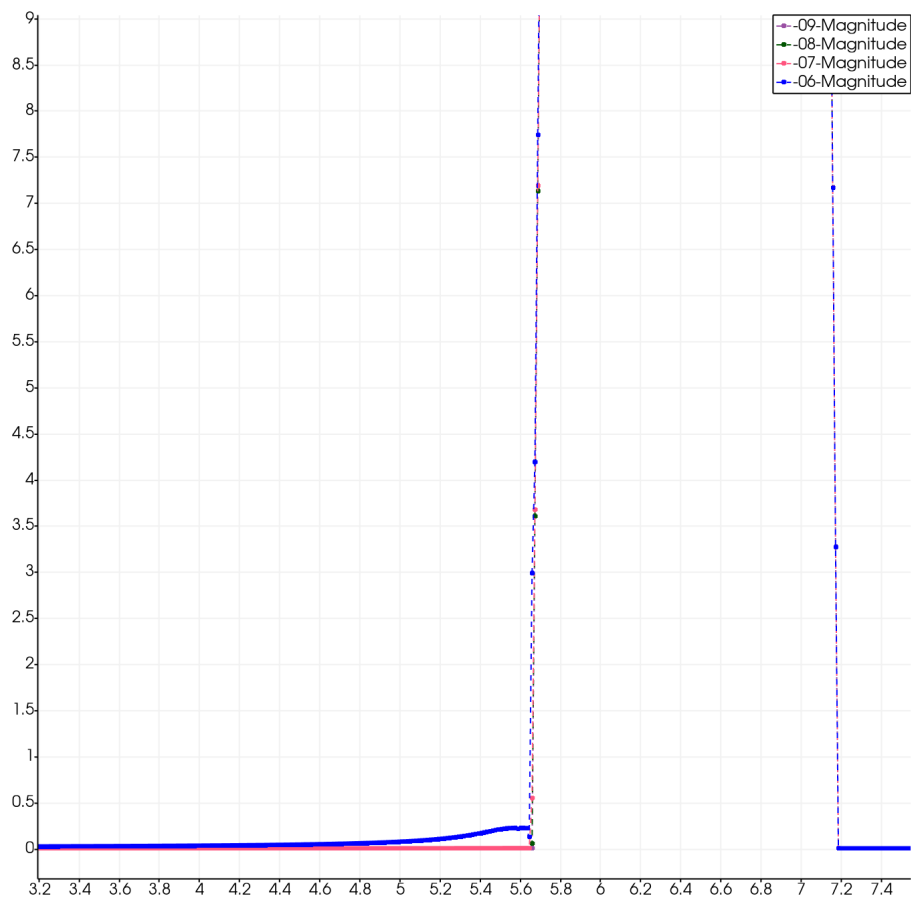
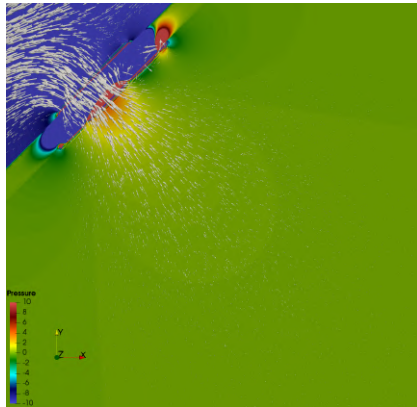
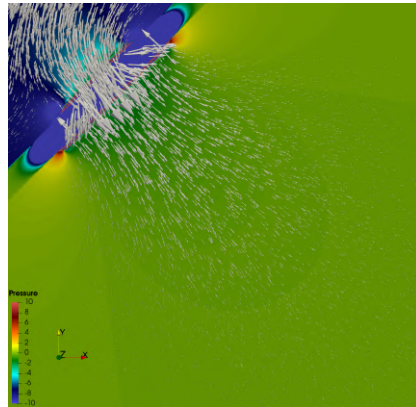


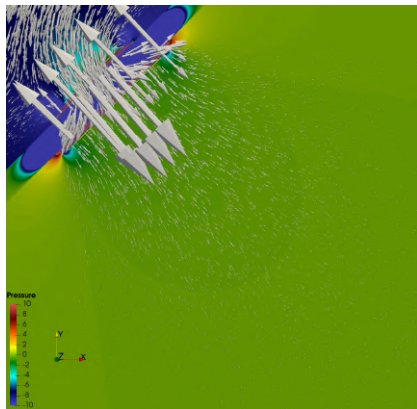
Figure 1.0.10: Plot over line for permeability: $1.36 \cdot 10^{-9} \mu m^2$ - $1.36 \cdot 10^{-6} \mu m^2$ in an open grid and $4 \mu m$ gap size.



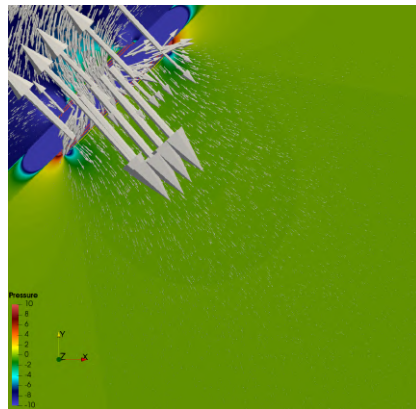
(a) Permeability: $1.36 \cdot 10^{-5} \mu m^2$



(b) Permeability: $1.36 \cdot 10^{-4} \mu m^2$



(c) Permeability: $1.36 \cdot 10^{-3} \mu m^2$



(d) Permeability: $1.36 \cdot 10^{-2} \mu m^2$

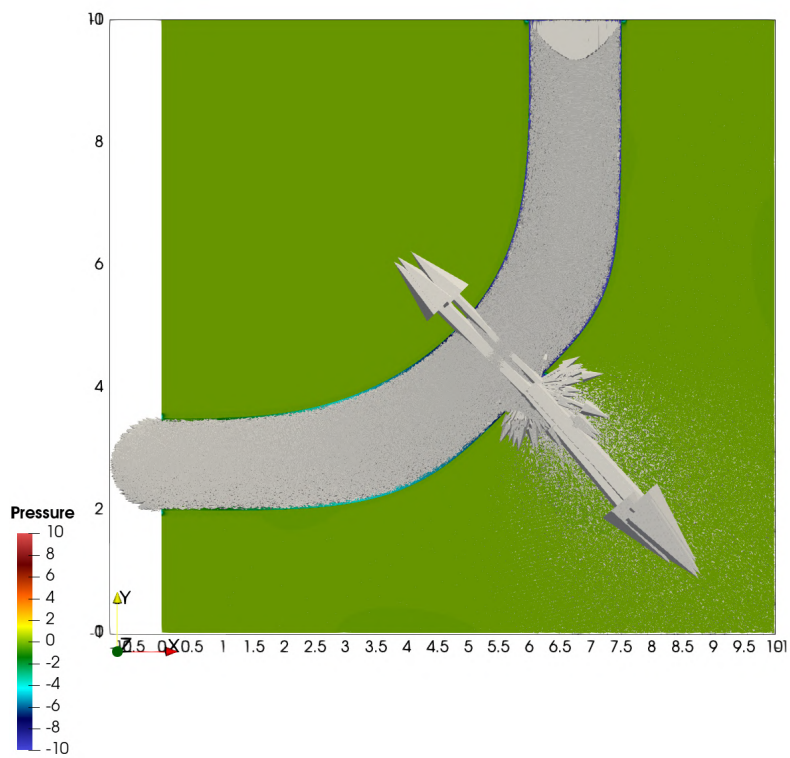


Figure 1.0.12: A better view of the simulation with permeability equal to $1.36 \cdot 10^{-2} \mu m^2$ in an open grid and $4 \mu m$ gap size.

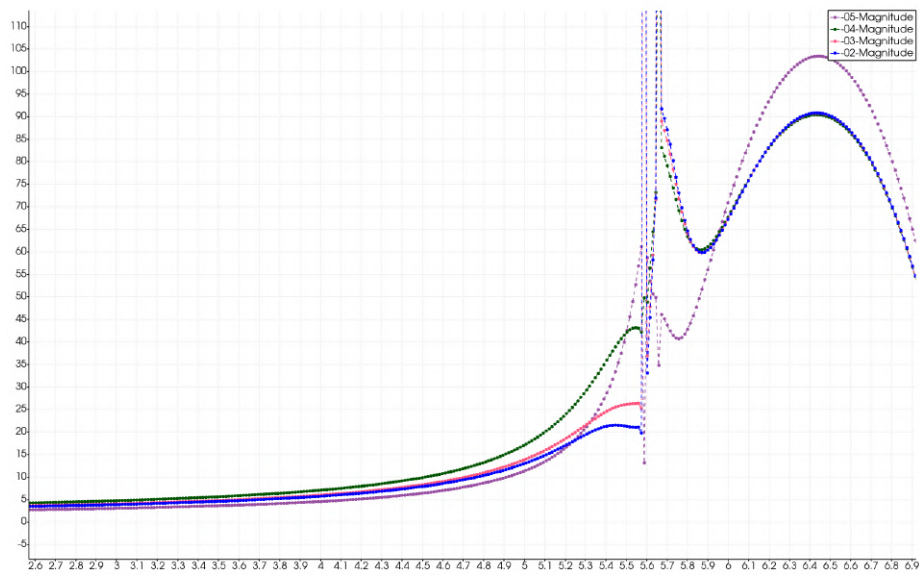


Figure 1.0.13: A zoomed view of plot over line for the simulation in an open grid and $4\mu m$ gap size. The maximum velocity magnitude that are outside the view are: $250 \frac{\mu}{s}$, $310 \frac{\mu}{s}$, $350 \frac{\mu}{s}$ and $410 \frac{\mu}{s}$ for the permeabilities $1.36 \cdot 10^{-5}$ - $1.36 \cdot 10^{-2}$, repectively.

Velocity and Pressure Development

The following figures illustrate the velocity and pressure development in an open grid and $4\mu\text{m}$ gap size, for each subdomain.

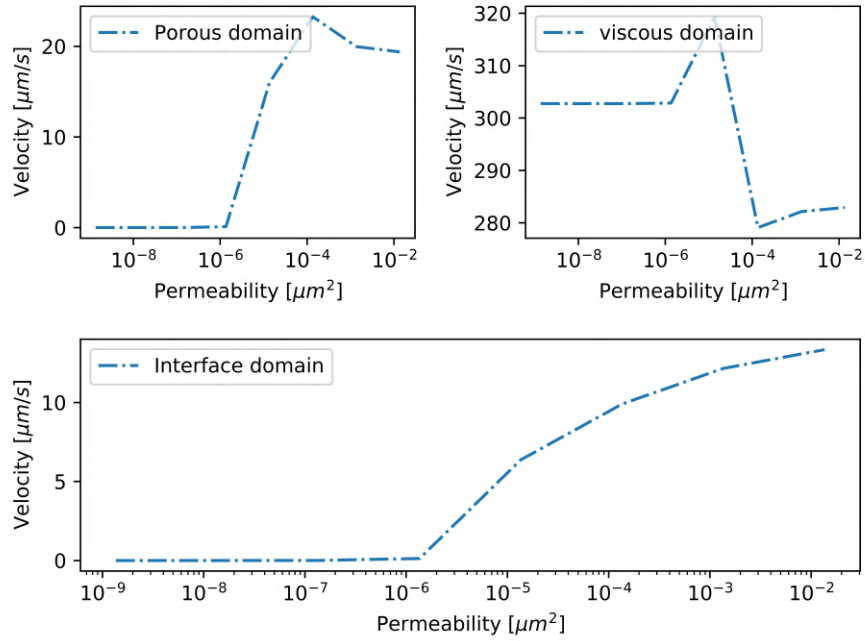


Figure 1.0.14: The velocity development for each subdomain in an open grid and $4\mu\text{m}$ gap size.

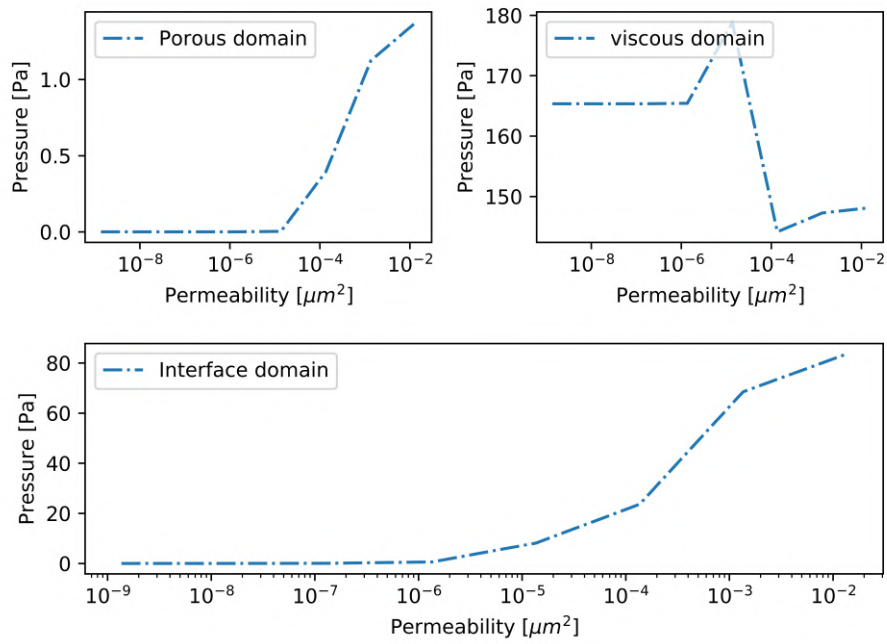


Figure 1.0.15: The pressure development for each subdomain in an open grid and $4\mu m$ gap size.

Time usage

The time used for the numerical experiments, where we have executed the code for each permeability separately:

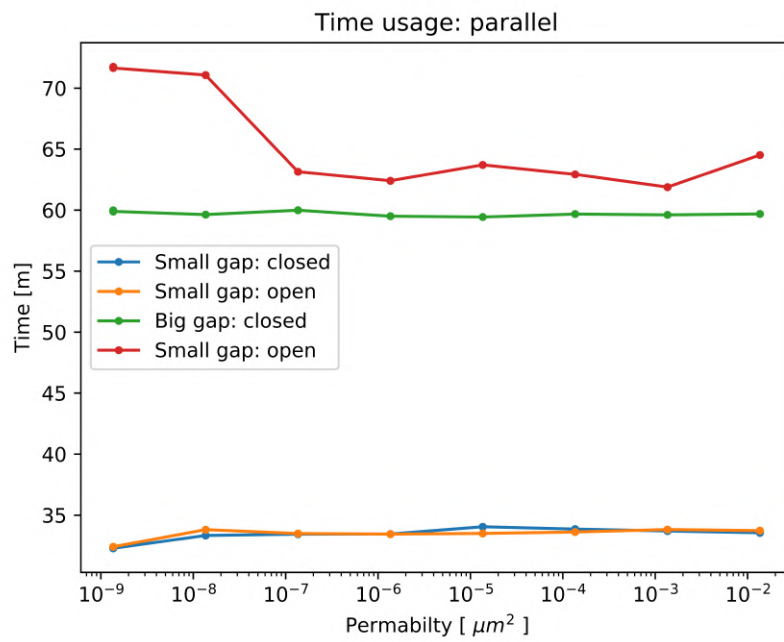


Figure 1.0.16: A view of time used against permeability for each experiments.

Bibliography

- [1] Lara Abouorm, Julien Bruchon, Nicolas Moulin, and Sylvain Drapier. Monolithic approach for coupling stokes-darcy in severe regimes. In *11e colloque national en calcul des structures*, Giens, France, May 2013. CSMA. URL <https://hal.archives-ouvertes.fr/hal-01722059>.
- [2] J. Gordon Betts, OpenStax College, BC Open Textbook Project, and BCcampus. Anatomy & physiology, 2013. URL <https://open.bccampus.ca/find-open-textbooks/?uuiid=f4873e49-e09c-469e-9ee8-9f14ca5a4e00>.
- [3] Wenbin Chen, Fang Wang, and Yanqiu Wang. Weak galerkin method for the coupled darcy-stokes flow. *IMA Journal of Numerical Analysis*, 36, 07 2014. doi: 10.1093/imanum/drv012.
- [4] Tzu-Chieh Chou, Juhyun Lee, Tzung K. Hsiai, and Yu-Chong Tai. A vacuum capillary viscometer that measures the viscosity of biofluids. *2017 19th International Conference on Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS)*, pages 1547–1550, 2017.
- [5] Christina Chun. What are the different types of tumor? URL <https://www.medicalnewstoday.com/articles/249141>.
- [6] Philippe G. Ciarlet. *The Finite Element Method for Elliptic Problems*. Society for Industrial and Applied Mathematics, 2002. doi: 10.1137/1.9780898719208. URL <https://epubs.siam.org/doi/abs/10.1137/1.9780898719208>.
- [7] Nils-Jørgen Knudsen Dal, Agnese Kocere, Jens Wohlmann, Simon Van Herck, Tobias A Bauer, Julien Resseguier, Shahla Bagherifam, Hilde Hyldmo, Matthias Barz, Bruno G De Geest, and Federico Fenaroli. Zebrafish embryos allow prediction of nanoparticle circulation times in mice and facilitate quantification of nanoparticle-cell interactions. *Small*, page e1906719, 2020.
- [8] Charles Patrick Davis. Cancer. URL <https://www.medicinenet.com/cancer/article.htm>.
- [9] Marco Discacciati and Alfio Quarteroni. Navier-stokes/darcy coupling: Modeling, analysis, and numerical approximation. *Revista Matemática Complutense*, 22, 07 2009. doi: 10.5209/rev_REMA.2009.v22.n2.16263.

- [10] Ida Norderhaug Drøsdal. Porous and viscous modeling of cerebrospinal fluid flow in the spinal canal associated with syringomyelia. URL <https://www.duo.uio.no/handle/10852/10872>.
- [11] Ada Johanne Ellingsrud. Preconditioning unified mixed discretizations of coupled darcy-stokes flow. URL <https://www.duo.uio.no/handle/10852/45338>.
- [12] Howard Elman, David Silvester, and Andrew Wathen. Finite elements and fast iterative solvers: With applications in incompressible fluid dynamics. 01 2006.
- [13] Lawrence C. Evans. *Partial differential equations*, volume 19. American Mathematical Society, 1998.
- [14] Gavin Fullstone, Jonathan Wood, M. Holcombe, and Giuseppe Battaglia. Modelling the transport of nanoparticles under blood flow using an agent-based approach. *Scientific reports*, 5:10649, 06 2015. doi: 10.1038/srep10649.
- [15] C. Geuzaine and J.-F. Remacle. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, 2009. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4676433/pdf/cgm-suppl.1-2015-115.pdf>.
- [16] Aniket V. Gore, Kathryn Monzo, Young R. Cha, Weijun Pan, and Brant M. Weinstein. Vascular development in the zebrafish. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3331685/pdf/cshperspectmed-ANG-a006684.pdf>.
- [17] Revolution Health and Wellness. The structure of artery wall. URL <http://www.revolutionhealth.org/arterial-remodeling-with-hypertension/>.
- [18] Kitware Inc. and Los Alamos National Laboratory. Paraview. URL <https://www.paraview.org/>.
- [19] Environmental Science Investigation. Circulation physiology. URL <https://esi.stanford.edu/circulation/circulation8.htm>.
- [20] Willi Jäger and Andro Mikelić. On the interface boundary condition of beavers, joseph, and saffman. *SIAM Journal on Applied Mathematics*, 60(4):1111–1127, 2000. ISSN 00361399. URL <http://www.jstor.org/stable/118544>.
- [21] Trygve Karper, Kent-Andre Mardal, and Ragnar Winther. Unified finite element discretizations of coupled darcy–stokes flow. *Numerical Methods for Partial Differential Equations*, 25(2):311–326, 2009. doi: 10.1002/num.20349. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/num.20349>.
- [22] Temesgen Kindo. Verify simulations with the method of manufactured solutions. URL <https://www.comsol.com/blogs/verify-simulations-with-the-method-of-manufactured-solutions/>.

- [23] Richard E. Klabunde. Hemodynamics (pressure, flow, and resistance). URL <https://www.cvphysiology.com/Hemodynamics/H001>.
- [24] Siew Lam, Yi Wu, Vinsensius Vega, Lance Miller, Jan Spitsbergen, Yan Tong, Huiqing Zhan, Kunde Govindarajan, Serene Lee, Sinnakarupan Mathavan, RK Murthy Karuturi, Donald Buhler, Edison Liu, and Zhiyuan Gong. Conservation of gene expression signatures between zebrafish and human liver tumors and tumor progression. *Nature biotechnology*, 24:73–5, 02 2006. doi: 10.1038/nbt1169.
- [25] H. P. Langtangen and K.-A. Mardal. *Introduction to Numerical Methods for Variational Problems*. 2016. URL <http://hplgit.github.io/fem-book/doc/pub/book/pdf/fem-book-4screen.pdf>.
- [26] William J. Layton, Friedhelm Schieweck, and Ivan Yotov. Coupling fluid flow with porous media flow. *SIAM J. Numer. Anal.*, 40:2003, 2003.
- [27] Juhyun Lee, Tzu-Chieh Chou, Dongyang Kang, Hanul Kang, Junjie Chen, Kyung-In Baek, Wei Wang, Yichen Ding, Dino Di Carlo, Yu-Chong Tai, and Tzung Hsiai. A rapid capillary-pressure driven micro-channel to demonstrate newtonian fluid behavior of zebrafish blood at high shear rates. *Scientific Reports*, 7, 05 2017. doi: 10.1038/s41598-017-02253-7.
- [28] Anders Logg. Dofin-convert. URL https://people.sc.fsu.edu/~jburkardt/py_src/dofin-convert/dofin-convert.html.
- [29] Anders Logg and Kent-Andre Mardal. Lectures on the finite element method. URL https://folk.uio.no/kent-and/sommerskole/material/book_20april.pdf.
- [30] Anders Logg, Kent-Andre Mardal, Garth N. Wells, et al. *Automated Solution of Differential Equations by the Finite Element Method*. Springer, 2012. ISBN 978-3-642-23098-1. doi: 10.1007/978-3-642-23099-8.
- [31] Flavien Loiseau. msh2xdmf. URL <https://github.com/floiseau/msh2xdmf>.
- [32] G. Majno, S. M. Shea, and M. Leventhal. Endothelial contraction induced by histamine-type mediators: an electron microscopic study. *The Journal of cell biology*, 42(3):647–672, Sep 1969. ISSN 0021-9525. doi: 10.1083/jcb.42.3.647. URL <https://doi.org/10.1083/jcb.42.3.647>.
- [33] Yu Matsumoto, Joseph W. Nichols, Kazuko Toh, Takahiro Nomoto, Horacio Cabral, Yutaka Miura, R. James Christie, Naoki Yamada, Tadayoshi Ogura, Mitsunobu R. Kano, Yasuhiro Matsumura, Nobuhiro Nishiyama, Tatsuya Yamasoba, You Han Bae, and Kazunori Kataoka. Vascular bursts enhance permeability of tumour blood vessels and improve nanoparticle delivery. *Nature Nanotechnology*, 11(6):533–538, Jun 2016. ISSN 1748-3395. doi: 10.1038/nnano.2015.342. URL <https://doi.org/10.1038/nnano.2015.342>.
- [34] Charles Molnar and Jane Gair. *Concepts of Biology-1st Canadian Edition*. BCcampus, 2015.

- [35] Dr Alan Mullan and Dr Aleksandra Marsh. Advantages of using zebrafish as a model organism. URL <https://andor.oxinst.com/learning/view/article/advantages-of-using-zebrafish-as-a-model-organism>.
- [36] S. P. Neuman. Theoretical derivation of darcy's law. *Acta Mechanica*, 25 (3):153–170, 1977. ISSN 1619-6937. doi: 10.1007/BF01376989.
- [37] William Louis Oberkampf, Charles Hirsch, and Timothy Guy Trucano. Verification, validation, and predictive capability in computational engineering and physics., 2003.
- [38] Theodoros G. Papaioannou and Christodoulos Stefanadis. Vascular wall shear stress: Basic principles and methods. URL https://www.hellenicjcardiol.org/archive/full_text/2005/1/2005_1_9.pdf.
- [39] Womersley J. R. Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known. *The Journal of physiology*, 127(3):553–563, Mar 1955. ISSN 0022-3751. doi: 10.1113/jphysiol.1955.sp005276. URL <https://pubmed.ncbi.nlm.nih.gov/14368548>. 14368548[pmid].
- [40] S. Rebay. Efficient unstructured mesh generation by means of delaunay triangulation and bowyer-watson algorithm. *Journal of Computational Physics; (United States)*, 106. doi: 10.1006/jcph.1993.1097.
- [41] Julien Resseguier. Julien region 1, 2019. URL https://drive.google.com/file/d/1uZRvbryPu7sZnYsDvGoT_1lA0fQ0hBH9/view.
- [42] Julien Resseguier. Julien region 2, 2019. URL <https://drive.google.com/file/d/1G2C1DP31xMMK-YXx0MoeFkg3nuU9shpd/view>.
- [43] Julien Resseguier. Julien region 3, 2019. URL https://drive.google.com/file/d/1CHo4Hm3EfNnYbFJL_nrGqeODTmhBps17/view.
- [44] Julien Resseguier. Julien region 4, 2019. URL <https://drive.google.com/file/d/1co5NinxKygYwCewub0L-4SW-ey7nY673/view>.
- [45] Seung-Sik Rho, Koji Ando, and Shigetomo Fukuhara. Dynamic regulation of vascular permeability by vascular endothelialcadherin-mediated endothelial cell-cell junctions. URL <https://www.nms.ac.jp/sh/jnms/2017/084040148.pdf>.
- [46] Béatrice Rivière and Ivan Yotov. Locally conservative coupling of stokes and darcy flows. *SIAM J. Numerical Analysis*, 42:1959–1977, 01 2005. doi: 10.1137/S0036142903427640.
- [47] Kambiz Salari and Patrick Knupp. *Code Verification by the Method of Manufactured Solutions*. 2000. URL <https://www.osti.gov/servlets/purl/759450/>.
- [48] Aamer Sandoo, Jet J. C. S. Veldhuijzen van Zanten, George S. Metsios, Douglas Carroll, and George D. Kitas. The endothelium and its role in regulating vascular tone. *The open cardiovascular medicine journal*, 4:302–312, Dec 2010. ISSN 1874-1924. doi: 10.2174/1874192401004010302. URL <https://pubmed.ncbi.nlm.nih.gov/21339899>. 21339899[pmid].

- [49] Caroline A. Schneider, Wayne S. Rasband, and Kevin W. Eliceiri. Nih image to imagej: 25 years of image analysis. *Nature Methods*, 9(7):671–675, Jul 2012. ISSN 1548-7105. doi: 10.1038/nmeth.2089. URL <https://doi.org/10.1038/nmeth.2089>.
- [50] Thomas N. Seyfried and Leanne C. Huysentruyt. On the origin of cancer metastasis. *Critical reviews in oncogenesis*, 18(1-2):43–73, 2013. ISSN 0893-9675. doi: 10.1615/critrevoncog.v18.i1-2.40. URL <https://pubmed.ncbi.nlm.nih.gov/23237552>.
- [51] V. M. Shapovalov. On the applicability of the ostwald–de waele model in solving applied problems. *Journal of Engineering Physics and Thermophysics*, 90(5):1213–1218, Sep 2017. ISSN 1573-871X. doi: 10.1007/s10891-017-1676-9. URL <https://doi.org/10.1007/s10891-017-1676-9>.
- [52] Joshua H. Smith and Joseph A.C. Humphrey. Interstitial transport and transvascular fluid exchange during infusion into brain and tumor tissue. *Microvascular Research*, 73(1):58–73, Jan 2007. ISSN 0026-2862. URL <http://www.sciencedirect.com/science/article/pii/S0026286206000896>.
- [53] Anthony J. Tavares Stefan Wilhelm et al. Analysis of nanoparticle delivery to tumours. URL <https://www.wilhelm-lab.com/wp-content/uploads/2020/01/natrevmats201614.pdf>.
- [54] Ms. T. Biology 11: Topic 22: The circulation system. URL <https://bilingualbiology11a.blogspot.com/2014/03/the-circulation-system.html>.
- [55] Dal NK Trousil J, Syrova Z and Rifampicin. Nanoformulation enhances treatment of tuberculosis in zebrafish. *Biomacromolecules*, 20, 2019. doi: 10.1021/acs.biomac.9b00214.
- [56] Petter Vollestad. Zebrafish report 2. https://gitlab.com/sorann/master/-/blob/master/Papers/Petter%20Vollestad/Zebrafish_20report2_2.pdf, 2019.
- [57] Petter Vollestad. Zebrafish report 4. https://gitlab.com/sorann/master/-/blob/master/Papers/Petter%20Vollestad/Zebrafish_20report4.pdf, 2019.
- [58] I. Bernard Weinstein and Kathleen Case. The history of cancer research: Introducing an aacr centennial series. URL <https://cancerres.aacrjournals.org/content/canres/68/17/6861.full.pdf>.
- [59] Frank M. White. *Viscous fluid flow*. McGraw-Hill Education, 2006.
- [60] Vega Wordpress. Fenics project. URL <https://fenicsproject.org>.
- [61] Nelson S. Yee, Natalia Ignatenko, Niklas Finnberg, Nikki Lee, and Douglas Stairs. Animal models of cancer biology. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4676433/pdf/cgm-suppl.1-2015-115.pdf>.